

用户手册

78K0/LE3

8 位单片微控制器

μPD78F0441	μPD78F0451	μPD78F0461
μPD78F0442	μPD78F0452	μPD78F0462
μPD78F0443	μPD78F0453	μPD78F0463
μPD78F0444	μPD78F0454	μPD78F0464
μPD78F0445	μPD78F0455	μPD78F0465

文档编号. U18696CA3V0UD00 (第 3 版)
发行日期 2009 年 3 月 NS

[备忘录]

CMOS 设备注意事项

① 输入引脚处的电压波形

输入噪音或一个反射波引起的波形失真可能导致错误发生。如果由于噪音等的影响使CMOS设备的输入电压范围保持在VIL(MAX)和VIH(MIN)之间，设备可能发生错误。在输入电平固定时以及输入电平从VIL (MAX)过渡到VIH (MIN)时的传输期间，要防止散射噪声影响设备。

② 未使用的输入引脚的处理

CMOS设备的输入端保持开路可能导致误操作。如果一个输入引脚未被连接，则由于噪音等原因可能会产生内部输入电平，从而导致误操作。CMOS设备的操作特性与Bipolar或NMOS设备不同。CMOS设备的输入电平必须借助上拉或下拉电路固定在高电平或低电平。每一个未使用引脚都应该通过附加电阻连接到VDD或GND。如果有可能尽量定义为输出引脚。对未使用引脚的处理因设备而异，必须遵循与设备相关的规定和说明。

③ ESD防护措施

如果MOS设备周围有强电场，将会击穿氧化栅极，从而影响设备的运行。因此必须采取措施，尽可能防止静电产生。一旦有静电，必须立即释放。对于环境必须有适当的控制。如果空气干燥，应当使用增湿器。建议避免使用容易产生静电的绝缘体。半导体设备的存放和运输必须使用抗静电容器、抗静电屏蔽袋或导电材料容器。所有的测试和测量工具包括工作台和工作面必须良好接地。操作员应当佩戴静电消除手带以保证良好接地。不能用手直接接触半导体设备。对于装配有半导体设备的PW板也应采取类似的静电防范措施。

④ 初始化之前的状态

在上电时MOS设备的初始状态是不确定的。在刚刚上电之后，具有复位功能的MOS设备并没有被初始化。因此上电不能保证输出引脚的电平，I/O设置和寄存器的内容。设备在收到复位信号后才进行初始化。具有复位功能的设备在上电后必须立即进行复位操作。

⑤ 电源开关顺序

在一个设备的内部操作和外部接口使用不同的电源的情况下，按照规定，应先在接通内部电源之后再接通外部电源。当关闭电源时，按照规定，先关闭外部电源再关闭内部电源。如果电源开关顺序颠倒，可能会导致设备的内部组件过电压，产生异常电流，从而引起内部组件的误操作和性能的退化。

对于每个设备电源的正确开关顺序必须依据设备的规范说明分别进行判断。

⑥ 电源关闭状态下的输入信号

不要向没有加电的设备输入信号或提供I/O上拉电源。因为输入信号或提供I/O上拉电源将引起电流注入，从而引起设备的误操作，并产生异常电流，从而使内部组件退化。

每个设备电源关闭时的信号输入必须依据设备的规范说明分别进行判断。

EEPROM 是 NEC 电子的注册商标。

SuperFlash®是 Silicon Storage Technology, Inc.的一个注册商标，已经在美国和日本等几个国家使用。

注意事项：该产品使用的SuperFlash®技术获得了**Silicon Storage Technology, Inc.**公司的授权。

- **本文档所登载的内容有效期截止至 2009 年 3 月，信息先于产品的生产周期发布。将来可能未经预先通知而更改。在实际进行生产设计时，请参阅各产品最新的数据表或数据手册等相关资料以获取本公司产品的最新规格。**
- 并非所有的产品和/或型号都向每个国家供应。请向本公司销售代表查询产品供应及其他信息。
- 未经本公司事先书面许可，禁止复制或转载本文件中的内容。否则因本文档所登载内容引发的错误，本公司概不负责。
- 本公司对于因使用本文件中列明的本公司产品而引起的，对第三者的专利、版权以及其它知识产权的侵权行为概不负责。本文件登载的内容不应视为本公司对本公司或其他人所有的专利、版权以及其它知识产权作出任何明示或默示的许可及授权。
- 本文件中的电路、软件以及相关信息仅用以说明半导体产品的运作和应用实例。用户如在设备设计中应用本文件中的电路、软件以及相关信息，应自行负责。对于用户或其他人因使用了上述电路、软件以及相关信息而引起的任何损失，本公司概不负责。
- 虽然本公司致力于提高半导体产品的质量及可靠性，但用户应同意并知晓，我们仍然无法完全消除出现产品缺陷的可能。为了最大限度地减少因本公司半导体产品故障而引起的对人身、财产造成损害（包括死亡）的危险，用户务必在其设计中采用必要的安全措施，如冗余度、防火和防故障等安全设计。
- 本公司产品质量分为：

  "标准等级"、"专业等级"以及"特殊等级"三种质量等级。

"特殊等级"仅适用于为特定用途而根据用户指定的质量保证程序所开发的日电电子产品。另外，各种日电电子产品的推荐用途取决于其质量等级，详见如下。用户在选用本公司的产品时，请事先确认产品的质量等级。

"标准等级"： 计算机，办公自动化设备，通信设备，测试和测量设备，音频·视频设备，家电，加工机械以及产业用机器人。

"专业等级"： 运输设备（汽车、火车、船舶等），交通用信号控制设备，防灾装置，防止犯罪装置，各种安全装置以及医疗设备（不包括专门为维持生命而设计的设备）。

"特殊等级： 航空器械，宇航设备，海底中继设备，原子能控制系统，为了维持生命的医疗设备、用于维持生命的装置或系统等。

除在本公司半导体产品的数据表或数据手册等资料中另有特别规定以外，本公司半导体产品的质量等级均为"标准等级"。如果用户希望在本公司设计意图以外使用本公司半导体产品，务必事先与本公司销售代表联系以确认本公司是否同意为该项应用提供支持。

（注）

（1） 本声明中的"本公司"是指日本电气电子株式会社（NEC Electronics Corporation）及其控股公司。

（2） 本声明中的"本公司产品"是指所有由日本电气电子株式会社开发或制造的产品或为日本电气电子株式会社（定义如上）开发或制造的产品。

M5 02.11-1

前言

读者

本手册旨在使用户了解 78K0/LE3 的功能及设计开发应用系统。
以下为目标设备产品。

78K0/LE3: μPD78F0441, 78F0442, 78F0443, 78F0444, 78F0445,
μPD78F0451, 78F0452, 78F0453, 78F0454, 78F0455,
μPD78F0461, 78F0462, 78F0463, 78F0464, 78F0465

目的

本手册旨使用户对功能描述的了解，以下为手册的**组织**结构。

组织

78K0/LE3 手册分为两部分: 本手册和指令部分 (与 78K0 微控制器通用).

78K0/LF3 **用户手册** **(本手册)**	**78K/0 系列指令用户手册**
• 引脚功能 • 内部模块功能 • 中断 • 片上外围功能 • 电气特性	• CPU 功能 • 指令设置 • 各指令说明

如何阅读本手册

在阅读本手册前，读者应掌握电子工程、逻辑电路和微控制器等电子工程方面的基础知识。

- 要掌握全部功能:
  → 请按**目录**顺序阅读本手册　标注“<R>” 表示主要修改部分。　修订部分可以很方便地通过在 PDF 文件中拷贝 “<R>” 并通过在 “查找:” 中指定来搜索查询。
- 如何解释寄存器格式：
  → 尖括号(<>)中的二进制位名称在 RA78K0 中被定义为保留字，并且在 CC78K0 中用#pragma sfr 指令定义为一个 sfr 变量.
- 要了解 78K0 微控制器指令功能的详细内容
  → 请参考 **78K/0 微控制器指令用户手册 (U12326E)**.

规则

数据规则: 数据的高位部分在左边，低位部分在右边

有效低电平表示法: $\overline{xxx}$ xxx (在引脚和信号名称上加划一条线)

注: 文中用**注**标注的相关术语的脚注

注意事项: 需要特别关注的信息

备注: 补充信息

数值的表示:
- 二进制 …×××× 或 ××××B
- 十进制 …××××
- 十六进制 …××××H

相关文档

本手册中提到的相关文档可能包括有初稿版本。 但是，初稿版本没有特别注明。

设备相关文档

文档名称	文档编号
78K0/LE3 用户手册	本手册
78K/0 系列指令用户手册	U12326E
78K0 微控制器自编程库 Type01 用户手册[注]	U18274E
78K0 微控制器 EEPROM™ 仿真库 Type01 用户手册[注]	U18275E

注: 本手册由工程部管理。 更多详细信息，请联系 NEC 电子销售代表。

开发工具相关文档 (软件) (用户手册)

文档名称		文档编号
RA78K0 Ver. 3.80 汇编包	操作篇	U17199E
	语言篇	U17198E
	结构化汇编语言篇	U17197E
CC78K0 Ver. 3.70 C 编译器	操作篇	U17201E
	语言篇	U17200E
ID78K0-QB Ver. 3.00 综合调试器	操作篇	U18492E
PM+ Ver. 6.30		U18416E

开发工具相关文档 (硬件) (用户手册)

文档名称	文档编号
QB-78K0LX3 片上仿真器	U18511E
QB-MINI2 具有编程功能的片上调试仿真器	U18371E

Flash存储器编程相关文档

文档名称	文档编号
PG-FP5 存储器编程器	U18865E

注意事项: 以上列出的相关文档可能会在无任何声明条件下修改。 开发设计时，请使用各文档的最新版本。

其他文档

文档名称	文档编号
半导体选择指南 - 产品和封装–	X13769X
半导体设备装配手册	注
NEC 半导体设备的质量等级	C11531E
NEC 半导体设备可靠性/质量控制系统	C10983E
半导体设备防静电 ESD 指南	C11892E

注: 可参阅“半导体设备装配手册”网站 (http：//www.necel.com/pkg/en/mount/index.html)。

注意事项: 以上列出的相关文档可能会在无任何声明条件下修改。 开发设计时，请使用各文档的最新版本。

目录

1.1 特征

- 指令最短执行时间可以在高速（0.2 μs：@高速系统时钟的操作频率为 10 MHz）和超低速（122 μs：@副系统时钟的操作频率为 32.768 kHz）之间改变
- 通用寄存器：8 位 × 32 个寄存器（8 位× 8 个寄存器× 4 组）
- ROM，RAM 的容量

产品型号 \ 项目	程序存储器（ROM）		数据存储器 内部高速 RAM [注1]	数据存储器 内部扩展 RAM [注1]	数据存储器 LCD 显示 RAM
μPD78F0441，78F0451，78F0461	Flash 存储器[注1]	16 KB	768 字节	–	<μPD78F044x，78F045x>
μPD78F0442，78F0452，78F0462		24 KB	1 KB		32 × 4 位（28× 8 位）
μPD78F0443，78F0453，78F0463		32 KB			[28 × 4 位（24× 8 位）][注2]
μPD78F0444，78F0454，78F0464		48 KB		1 KB	<μPD78F046x>
μPD78F0445，78F0455，78F0465		60 KB			24 × 4 位（20 × 8 位）
					[20 × 4 位（16 × 8 位）][注2]

注 **1. 通过内部存储器容量切换寄存器（IMS）和内部扩展 RAM 容量切换寄存器（IXS），可以改变内部 Flash 存储器、内部高速 RAM 的容量和内部扩展 RAM 的容量。**

<R> **2. 圆括号中的项目在 8com 时适用。**
方括号中的项目在使用底端的 UART6 引脚（RxD6，TxD6）时适用。

- 内置单电源 Flash 存储器
- 自编程（具有引导交换功能）
- 片上调试功能
- 内置上电清零 （POC） 电路和低电压检测器（LVI）
- 内置看门狗定时器（在内部低速振荡时钟下可操作）
- LCD 控制器/驱动器（外部电阻分压和内部电阻分压可切换使用）

μPD78F044x： Segment 信号：28，公共端信号： 8（1/4 偏压）
：Segment 信号：32，公共端信号： 4（1/3 偏压）
：Segment 信号：32，公共端信号： 3（1/3，1/2 偏压）
：Segment 信号：32，公共端信号： 2（1/2 偏压）
：Segment 信号：32，公共端信号： 1（静态）

μPD78F045x： Segment 信号：28，公共端信号： 8（1/4 偏压）
：Segment 信号：32，公共端信号： 4（1/3 偏压）
：Segment 信号：32，公共端信号： 3（1/3，1/2 偏压）
：Segment 信号：32，公共端信号： 2（1/2 偏压）
：Segment 信号：32，公共端信号： 1（静态）

μPD78F046x： Segment 信号：20，公共端信号： 8（1/4 偏压）
：Segment 信号：24，公共端信号： 4（1/3 偏压）
：Segment 信号：24，公共端信号： 3（1/3，1/2 偏压）
：Segment 信号：24，公共端信号： 2（1/2 偏压）
：Segment 信号：24，公共端信号： 1（静态）

<R> ○ 片上 segment 键扫描功能：8 通道

○ 片上按键中断功能：5 通道

○ 片上蜂鸣器输出控制器

○ I/O 端口：46

○ 定时器： 9 通道

- 16 位定时器/事件计数器： 1 通道
- 8 位定时器/事件计数器： 3 通道
- 8 位定时器： 3 通道
- 实时计数器（RTC）： 1 通道
- 看门狗定时器： 1 通道

○ 串行接口： 2 通道

- UART （LIN（本地内联网）总线支持）： 1 通道
- CSI/UART [注]： 1 通道

注 可选择复用功能引脚的任意一种功能。

○ 16 位 ΔΣ 型 A/D 转换器[注]： 3 通道（仅限μPD78F046x）

○ 10 位逐次逼近型 A/D 转换器： 8 通道（仅限μPD78F045x 和 78F046x）

○ 遥控接收器

○ 曼彻斯特编码发生器

○ 电源电压： V_{DD} = 1.8 至 5.5 V

○ 工作环境温度： T_A = –40 至 +85°C

注 16 位ΔΣ型 A/D 转换器的规格可能已经更改。
详细规格，请联系 NEC 电子销售代表或授权的经销商。

1.2 应用

数码相机，AV 设备，家用电器，仪表，卫生保健设备，和测量设备等。

1.3 订购信息

- **Flash** 存储器版本 （无铅产品）

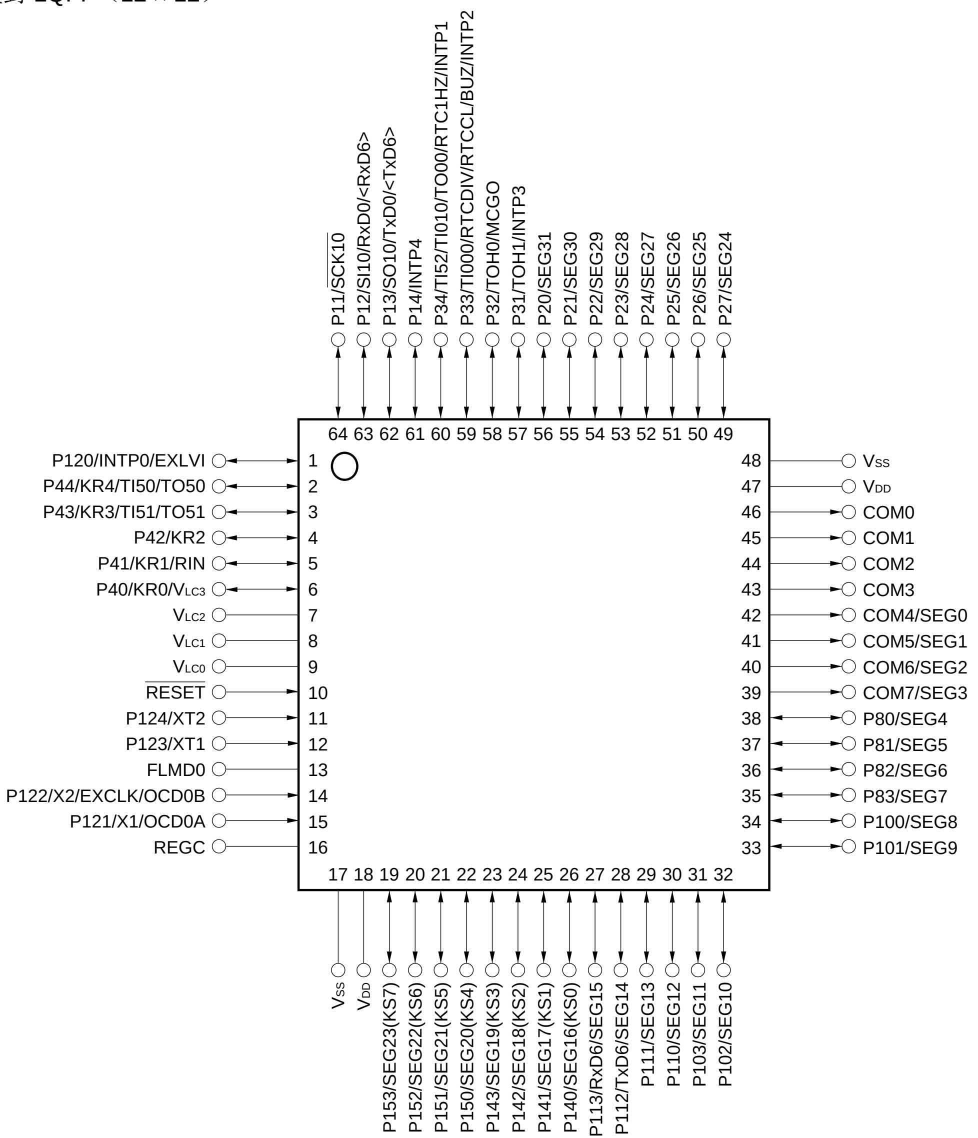
型 号	封 装
μPD78F0441GB-GAH-AX	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0442GB-GAH-AX	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0443GB-GAH-AX	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0444GB-GAH-AX	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0445GB-GAH-AX	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0441GK-GAJ-AX	64 脚塑封 LQFP（12 × 12）
μPD78F0442GK-GAJ-AX	64 脚塑封 LQFP（12 × 12）
μPD78F0443GK-GAJ-AX	64 脚塑封 LQFP（12 × 12）
μPD78F0444GK-GAJ-AX	64 脚塑封 LQFP（12 × 12）
μPD78F0445GK-GAJ-AX	64 脚塑封 LQFP（12 × 12）
μPD78F0451GB-GAH-AX	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0452GB-GAH-AX	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0453GB-GAH-AX	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0454GB-GAH-AX	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0455GB-GAH-AX	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0451GK-GAJ-AX	64 脚塑封 LQFP（12 × 12）
μPD78F0452GK-GAJ-AX	64 脚塑封 LQFP（12 × 12）
μPD78F0453GK-GAJ-AX	64 脚塑封 LQFP（12 × 12）
μPD78F0454GK-GAJ-AX	64 脚塑封 LQFP（12 × 12）
μPD78F0455GK-GAJ-AX	64 脚塑封 LQFP（12 × 12）
μPD78F0461GB-GAH-AX [注]	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0462GB-GAH-AX [注]	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0463GB-GAH-AX [注]	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0464GB-GAH-AX [注]	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0465GB-GAH-AX [注]	64 脚塑封 LQFP（密间距）（10 × 10）
μPD78F0461GK-GAJ-AX [注]	64 脚塑封 LQFP（12 × 12）
μPD78F0462GK-GAJ-AX [注]	64 脚塑封 LQFP（12 × 12）
μPD78F0463GK-GAJ-AX [注]	64 脚塑封 LQFP（12 × 12）
μPD78F0464GK-GAJ-AX [注]	64 脚塑封 LQFP（12 × 12）
μPD78F0465GK-GAJ-AX [注]	64 脚塑封 LQFP（12 × 12）

注 开发中

1.4 引脚配置（俯视图）

（1） μPD78F0441，78F0442，78F0443，78F0444，78F0445

- 64 脚塑封 LQFP（密间距）（10 × 10）
- 64 脚塑封 LQFP（12 × 12）



注意事项 **1.** 通过一个电容（**0.47** 至 **1 μF**：推荐值）将 **REGC** 引脚连接到 V_{SS}。

2. 用 **flash** 存储器编程器写入时，只能使用 **UART6** 功能（**RxD6** 和 **TxD6**）的底部引脚（引脚编号 **27** 和 **28**），不能通过顶部引脚（引脚编号 **63** 和 **62**）执行写入。

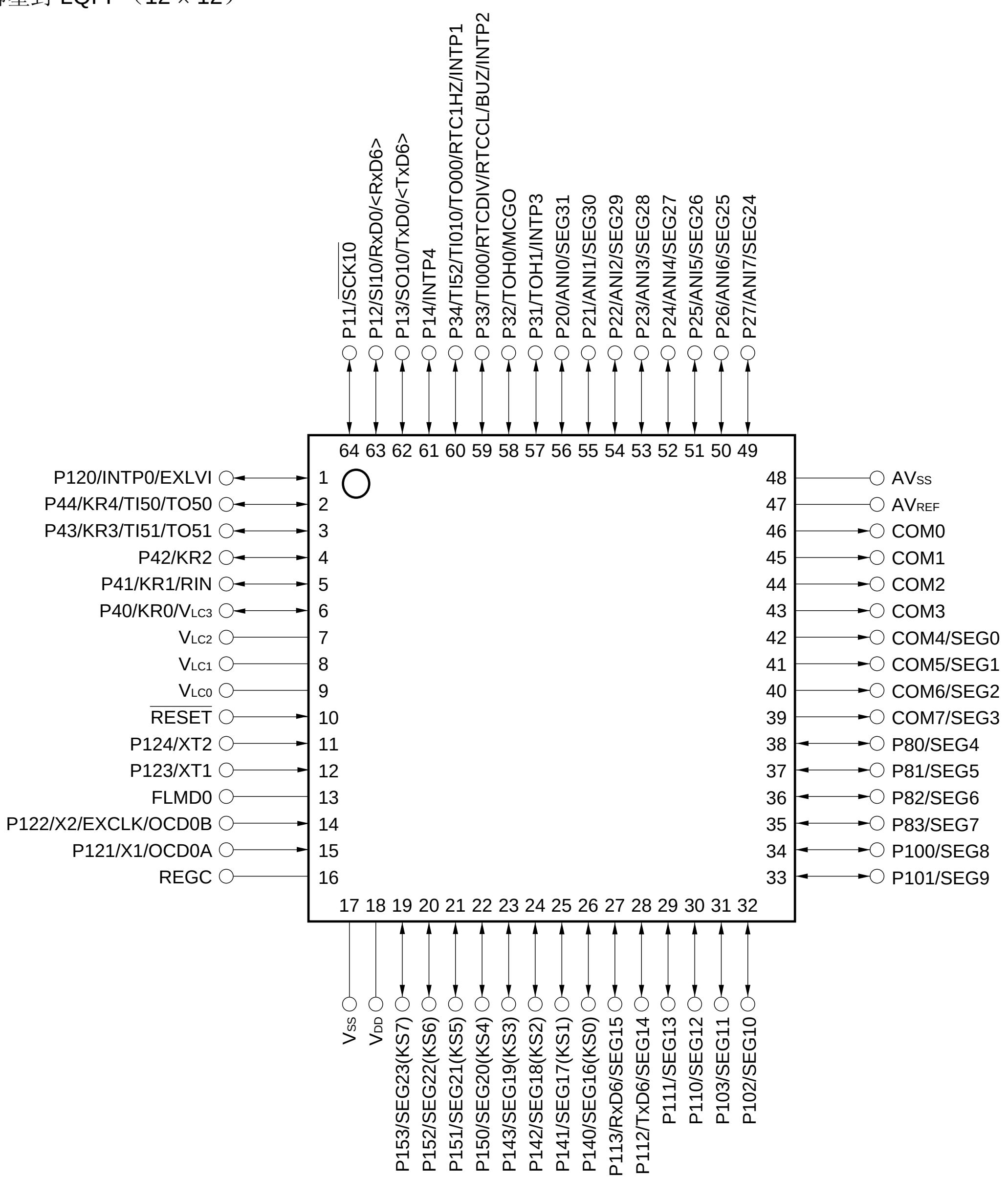
3. 应该使 V_{DD}（引脚编号 **18**）和 V_{DD}（引脚编号 **47**），V_{SS}（引脚编号 **17**）和 V_{SS}（引脚编号 **48**）保持相同的电位。

备注 **1.** 尖括号（< >）内的功能可以通过输入转换控制寄存器（ISC）来分配。

<R> **2.** 圆括号中的功能可以通过设置 LCD 模式寄存器（LCDMD）来使用。

（2） **μPD78F0451，78F0452，78F0453，78F0454，78F0455**

- 64 脚塑封 LQFP（密间距）（10 × 10）
- 64 脚塑封 LQFP（12 × 12）



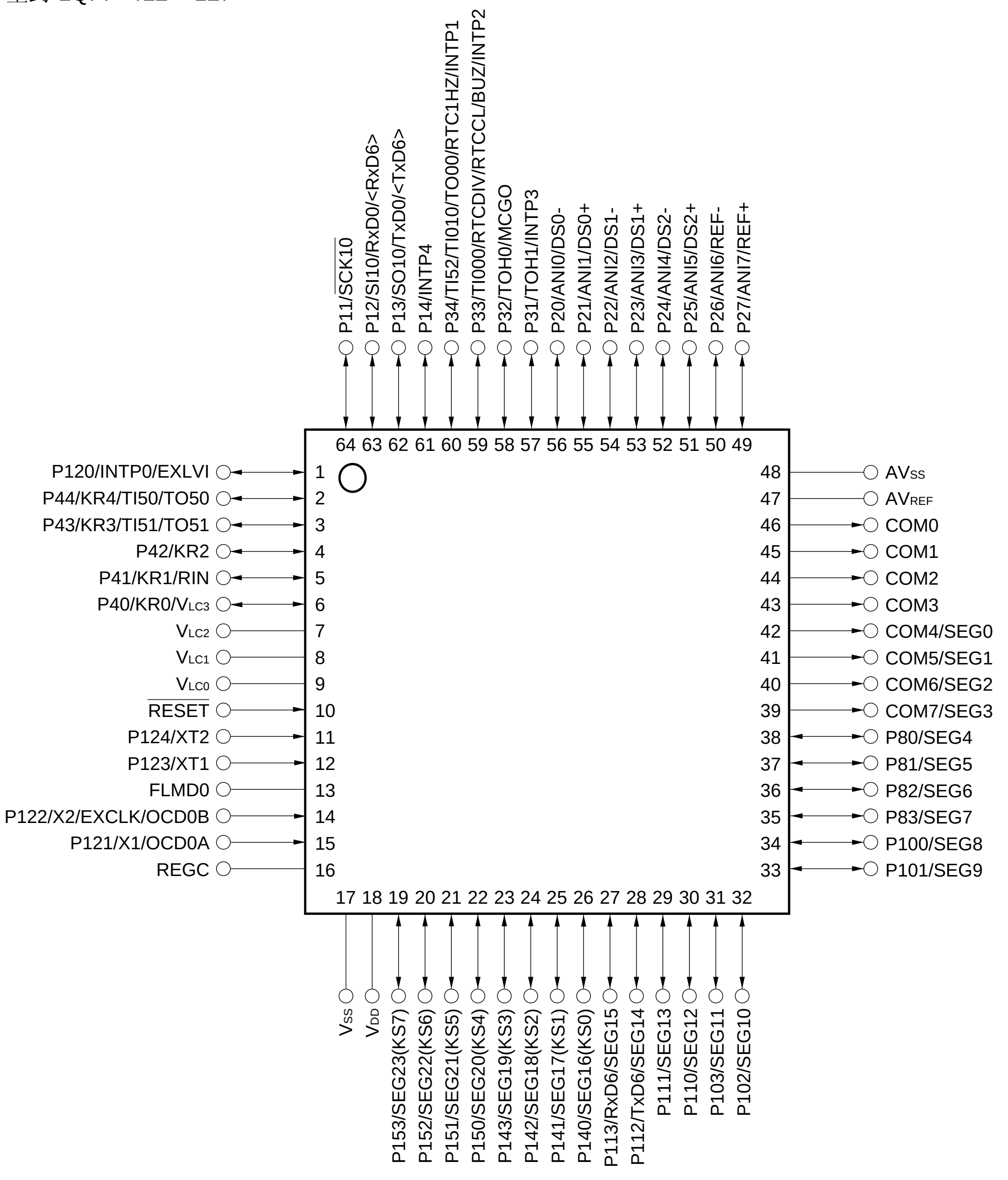
注意事项
1. **AV$_{SS}$ 与连接到 V$_{SS}$。**
2. **通过一个电容（0.47 至 1 μF：推荐值）将 REGC 引脚连接到 V$_{SS}$。**
3. **复位释放后，ANI0/P20 至 ANI7/P27 被设置为模拟输入模式。**
4. **用 flash 存储器编程器写入时，只能使用 UART6 功能（RxD6 和 TxD6）的底部引脚（引脚编号 27 和 28），不能通过顶部引脚（引脚编号 63 和 62）执行写入。**

备注
1. 尖括号（< >）内的功能可以通过输入转换控制寄存器（ISC）来分配。

<R> 2. 圆括号中的功能可以通过设置 LCD 模式寄存器（LCDMD）来使用。

（3） μPD78F0461，78F0462，78F0463，78F0464，78F0465

- 64 脚塑封 LQFP（密间距）（10 × 10）
- 64 脚塑封 LQFP（12 × 12）



注意事项 **1. AVss 与连接到 Vss。**

2. 通过一个电容（**0.47** 至 **1** μ**F**：推荐值）将 **REGC** 引脚连接到 **Vss**。

3. 复位释放后，**ANI0/P20** 至 **ANI7/P27** 被设置为模拟输入模式。

4. 用 **flash** 存储器编程器写入时，只能使用 **UART6** 功能（**RxD6** 和 **TxD6**）的底部引脚（引脚编号 **27** 和 **28**），不能通过顶部引脚（引脚编号 **63** 和 **62**）执行写入。

备注 **1.** 尖括号（< >）内的功能可以通过输入转换控制寄存器（ISC）来分配。

<R> **2.** 圆括号中的功能可以通过设置 LCD 模式寄存器（LCDMD）来使用。

引脚标识

ANI0 至 ANI7 [注 1]：	模拟输入	REF+ [注 2]：	ΔΣ模拟参考电压（+）
AV_{REF} [注 1]：	模拟参考电压	REF– [注 2]：	ΔΣ模拟参考电压（–）
AV_{SS} [注 1]：	模拟地	RIN：	遥控输入
BUZ：	蜂鸣器输出	RTC1HZ：	实时计数器修正时钟（1 Hz） 输出
DS0+ 至 DS2+ [注 2]：	ΔΣ 模拟输入（+）	RTCCL：	实时计数器时钟（32.768 kHz 振荡） 输出
DS0– 至 DS2– [注 2]：	ΔΣ模拟输入（–）	RTCDIV：	实时计数器时钟（32.768 kHz 分频） 输出
EXCLK：	外部时钟输入（主系统时钟）	SEG0 至 SEG23：	Segment 输出
EXLVI：	外部电压输入 用于低电压检测器	SEG24 至 SEG31 [注 3]：	Segment 输出
FLMD0：	Flash 编程模式	SEG16（KS0） 至 SEG23（KS7）:	Segment 键扫描
INTP0 至 INTP4：	外部中断输入	$\overline{SCK10}$：	串行时钟输入 /输出
KR0 至 KR4：	按键返回	SI10：	串行数据输入
MCGO：	曼彻斯特编码发生器输出	SO10：	串行数据输出
OCD0A，OCD0B：	片上调试输入/输出	TI000，TI010：	定时器输入
P11 至 P14：	端口 1	TI50，TI51，TI52：	定时器输入
P20 至 P27：	端口 2	TO00：	定时器输出
P31 至 P34：	端口 3	TO50，TO51：	定时器输出
P40 至 P44：	端口 4	TOH0，TOH1：	定时器输出
P80 至 P83：	端口 8	TxD0，TxD6：	发送数据
P100 至 P103：	端口 10	V_{DD}：	电源电压
P110 至 P113：	端口 11	V_{SS}：	地
P120 至 P124：	端口 12	V_{LC0} 至 V_{LC3}：	LCD 电源
P140 至 P143：	端口 14	X1，X2：	晶体振荡器（主系统时钟）
P150 至 P153：	端口 15	XT1，XT2：	晶体振荡器（副系统时钟）
REGC	稳压电容		
$\overline{RESET}$：	复位		
RxD0，RxD6：	接收数据		

注 **1.** 仅限μPD78F045x 和 78F046x。
2. 仅限μPD78F046x。
3. 仅限μPD78F044x 和 78F045x。

1.5 78K0/Lx3 微控制器系列介绍

ROM	RAM	78K0/LC3	78K0/LD3	78K0/LE3	78K0/LF3
		48 脚	52 脚	64 脚	80 脚
60 KB	2 KB	–	–	μPD78F0465 μPD78F0455 μPD78F0445	μPD78F0495 μPD78F0485 μPD78F0475
48 KB	2 KB	–	–	μPD78F0464 μPD78F0454 μPD78F0444	μPD78F0494 μPD78F0484 μPD78F0474
32 KB	1 KB	μPD78F0413 μPD78F0403	μPD78F0433 μPD78F0423	μPD78F0463 μPD78F0453 μPD78F0443	μPD78F0493 μPD78F0483 μPD78F0473
24 KB	1 KB	μPD78F0412 μPD78F0402	μPD78F0432 μPD78F0422	μPD78F0462 μPD78F0452 μPD78F0442	μPD78F0492 μPD78F0482 μPD78F0472
16 KB	768 B	μPD78F0411 μPD78F0401	μPD78F0431 μPD78F0421	μPD78F0461 μPD78F0451 μPD78F0441	μPD78F0491 μPD78F0481 μPD78F0471
8 KB	512 B	μPD78F0410 μPD78F0400	μPD78F0430 μPD78F0420	–	–

78K0/Lx3 微控制器的功能在下表列出 （1/3）

项目 \ 型号	78K0/LC3								78K0/LD3							
	μPD78F040x				μPD78F041x				μPD78F042x				μPD78F043x			
	48 脚								52 脚							
Flash 存储器（KB）	8	16	24	32	8	16	24	32	8	16	24	32	8	16	24	32
RAM （KB）	0.5	0.75	1	1	0.5	0.75	1	1	0.5	0.75	1	1	0.5	0.75	1	1
电源电压	V_{DD} = 1.8 至 5.5 V															
稳压器	提供															
指令最短执行时间	0.2 μs （10 MHz： V_{DD} = 2.7 至 5.5 V）/ 0.4 μs （5 MHz： V_{DD} = 1.8 至 5.5 V）															
时钟 主时钟 高速系统时钟	10 MHz： V_{DD} = 2.7 至 5.5 V/5 MHz： V_{DD} = 1.8 至 5.5 V															
时钟 主时钟 内部高速振荡时钟	8 MHz （典型值）： V_{DD} = 1.8 至 5.5 V															
时钟 副时钟	32.768 kHz （典型值）： V_{DD} = 1.8 至 5.5 V															
时钟 内部低速振荡时钟	240 kHz （典型值）： V_{DD} = 1.8 至 5.5 V															
端口 总计	30								34							
定时器 16 位（TM0）	1 通道															
定时器 8 位（TM5）	3 通道															
定时器 8 位（TMH）	3 通道															
定时器 RTC	1 通道															
定时器 WDT	1 通道															
串行接口 3 线 CSI	–								1 通道[注 1]							
串行接口 UART	1 通道								1 通道[注 1]							
串行接口 支持 LIN 总线的 UART	1 通道[注 2]								1 通道[注 3]							
LCD 类型	外部电阻分压和内部电阻分压可切换使用。															
LCD Segment 信号	22（18） [20（16）][注 4, 5]								24（20） [21（17）][注 4, 5]							
LCD 公共端信号	4 （8）[注 4]															
10 位逐次逼近型 A/D	–				6 通道				–				6 通道			
16 位 ΔΣ 型 A/D	–															
中断 外部	5															
中断 内部	17				18				19				20			
Segment 键源信号输出	8 通道								8 通道							
按键中断	3 通道								5 通道							
复位 $\overline{\text{RESET}}$ 引脚	提供															
复位 POC	1.59 V ± 0.15 V （上升到 1.8 V 所用时间 ： 3.6 ms （最大值））															
复位 LVI	可选择 16 个电源电压检测级别															
复位 WDT	提供															
时钟输出	–															
蜂鸣器输出	提供															
遥控接收器	–								提供							
MCG	提供															
片上调试功能	提供															
工作环境温度	T_A = –40 至 +85°C															

注 **1.** 因为 3 线 CSI 和 UART 是复用功能引脚，所以必须指定使用其中一个功能。
2. 支持 LIN 总线的 UART 引脚可以更改为 UART 引脚（引脚编号 47 和 48）。
3. 支持 LIN 总线的 UART 引脚可以更改为 3 线 CSI/UART 引脚（引脚编号 50 和 51）。
4. 括号内的值是使用 8com 时的信号输出数。
<R> **5.** 方括号中的值是使用底部 UART6 引脚（RxD6，TxD6）时的信号输出数量。

（2/3）

项目＼型号	78K0/LE3														
	μPD78F044x					μPD78F045x					μPD78F046x				
	64 脚														
Flash 存储器（KB）	16	24	32	48	60	16	24	32	48	60	16	24	32	48	60
RAM （KB）	0.75	1	1	2	2	0.75	1	1	2	2	0.75	1	1	2	2
电源电压	V_{DD} = 1.8 至 5.5 V														
稳压器	提供														
指令最短执行时间	0.2 μs （10 MHz： V_{DD} = 2.7 至 5.5 V）/ 0.4 μs （5 MHz： V_{DD} = 1.8 至 5.5 V）														
时钟 主时钟 高速系统时钟	10 MHz： V_{DD} = 2.7 至 5.5 V/5 MHz： V_{DD} = 1.8 至 5.5 V														
时钟 主时钟 内部高速振荡时钟	8 MHz （典型值）： V_{DD} = 1.8 至 5.5 V														
时钟 副时钟	32.768 kHz （典型值）： V_{DD} = 1.8 至 5.5 V														
时钟 内部低速振荡时钟	240 kHz （典型值）： V_{DD} = 1.8 至 5.5 V														
端口 总计	46														
定时器 16 位 （TM0）	1 通道														
定时器 8 位（TM5）	3 通道														
定时器 8 位（TMH）	3 通道														
定时器 RTC	1 通道														
定时器 WDT	1 通道														
串行接口 3 线 CSI/UART 注 1	1 通道														
串行接口 支持 LIN 总线的 UART 注 2	1 通道														
LCD 类型	外部电阻分压和内部电阻分压可切换使用。														
LCD Segment 信号	32（28） [28（24）] 注 3，4										24（20） [20（16）] 注 3，4				
LCD 公共端信号	4 （8） 注 3														
10 位逐次逼近型 A/D	–					8 通道									
16 位 ΔΣ 型 A/D	–										3 通道				
中断 外部	6														
中断 内部	19					20					21				
Segment 键源信号输出	8 通道														
按键中断	5 通道														
复位 $\overline{\text{RESET}}$ 引脚	提供														
复位 POC	1.59 V ±0.15 V （上升到 1.8 V 所用时间 ：3.6 ms （最大值））														
复位 LVI	可选择 16 个电源电压检测级别														
复位 WDT	提供														
时钟输出	–														
蜂鸣器输出	提供														
遥控接收器	提供														
MCG	提供														
片上调试功能	提供														
工作环境温度	T_A = –40 至 +85°C														

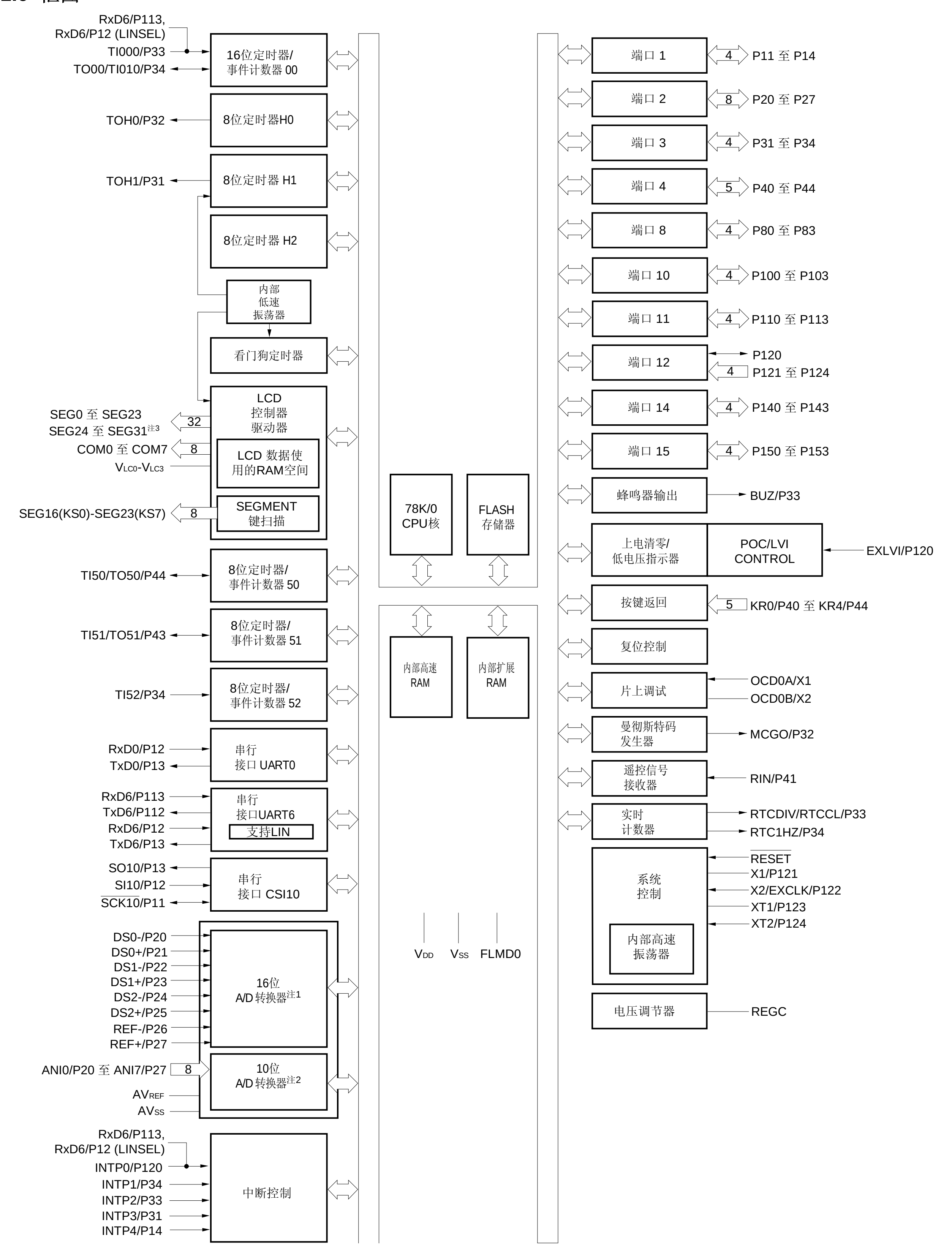
注 **1.** 在复用功能引脚中可以任意选择一种功能。
2. 支持 LIN 总线的 UART 引脚可以更改为 3 线 CSI/UART 引脚（引脚编号 62 和 63）。
3. 括号内的值是使用 8com 时的信号输出数。
<R> **4.** 方括号中的值是使用底部 UART6 引脚（RxD6，TxD6）时的信号输出数量。

（3/3）

项目 \ 型号	78K0/LF3														
	μPD78F047x					μPD78F048x					μPD78F049x				
	80 脚														
Flash 存储器（KB）	16	24	32	48	60	16	24	32	48	60	16	24	32	48	60
RAM （KB）	0.75	1	1	2	2	0.75	1	1	2	2	0.75	1	1	2	2
电源电压	V_{DD} = 1.8 至 5.5 V														
稳压器	提供														
指令最短执行时间	0.2 μs （10 MHz： V_{DD} = 2.7 至 5.5 V）/ 0.4 μs （5 MHz： V_{DD} = 1.8 至 5.5 V）														
时钟 主时钟 高速系统时钟	10 MHz： V_{DD} = 2.7 至 5.5 V/5 MHz： V_{DD} = 1.8 至 5.5 V														
时钟 主时钟 内部高速振荡时钟	8 MHz （典型值）： V_{DD} = 1.8 至 5.5 V														
时钟 副时钟	32.768 kHz （典型值）： V_{DD} = 1.8 至 5.5 V														
时钟 内部低速振荡时钟	240 kHz （典型值）： V_{DD} = 1.8 至 5.5 V														
端口 总计	62														
定时器 16 位 （TM0）	1 通道														
定时器 8 位（TM5）	3 通道														
定时器 8 位（TMH）	3 通道														
定时器 RTC	1 通道														
定时器 WDT	1 通道														
串行接口 3 线 CSI/UART [注 1]	1 通道														
串行接口 自动发送发送/接收 3 线 CSI	1 通道														
串行接口 支持 LIN 总线的 UART [注 2]	1 通道														
LCD 类型	外部电阻分压和内部电阻分压可切换使用。														
LCD Segment 信号	40（36） [36（32）] [注 3，4]										32（28） [28（24）] [注 3，4]				
LCD 公共端信号	4 （8） [注 3]														
10 位逐次逼近型 A/D	–					8 通道									
16 位 ΔΣ 型 A/D	–										3 通道				
中断 外部	7														
中断 内部	20					21					22				
Segment 键源信号输出	8 通道														
按键中断	8 通道														
复位 $\overline{\text{RESET}}$ 引脚	提供														
复位 POC	1.59 V ±0.15 V （上升到 1.8 V 所用时间： 3.6 ms （最大值））														
复位 LVI	可选择 16 个电源电压检测级别														
复位 WDT	提供														
时钟输出/蜂鸣器输出	提供														
遥控接收器	提供														
MCG	提供														
片上调试功能	提供														
工作环境温度	T_A = –40 至 +85°C														

注　**1.** 在复用功能引脚中可以任意选择一种功能。
2. 支持 LIN 总线的 UART 引脚可以更改为 3 线 CSI/UART 引脚（引脚编号 75 和 76）。
3. 括号内的值是使用 8com 时的信号输出数。
<R> **4.** 方括号中的值是使用底部 UART6 引脚（RxD6，TxD6）时的信号输出数量。

1.6 框图



注　**1.** 仅限μPD78F046x 。

2. 仅限μPD78F045x 和 78F046x 。

3. 仅限μPD78F044x 和 78F045x 。

1.7 功能概述（μPD78F044x）

（1/2）

项目		μPD78F0441	μPD78F0442	μPD78F0443	μPD78F0444	μPD78F0445
内部存储器	Flash 存储器（支持自编程）[注]	16 KB	24 KB	32 KB	48 KB	60 KB
	高速 RAM [注]	768 字节	1 KB			
	扩展 RAM [注]	–			1 KB	
	LCD 显示 RAM	32 × 4 位 （4 com） 或 28 × 8 位 （8 com）				
存储空间		64 KB				
主系统时钟（振荡频率）	高速系统时钟	X1 （晶体/陶瓷）振荡器，外部主系统时钟输入（EXCLK） 2 至 10 MHz：V_{DD} = 2.7 至 5.5 V， 2 至 5 MHz： V_{DD} = 1.8 至 5.5 V				
	内部高速振荡时钟	内部振荡器 8 MHz （典型值）： V_{DD} = 1.8 至 5.5 V				
副系统时钟（振荡频率）		XT1 （晶体） 振荡器 32.768 kHz （典型值）： V_{DD} = 1.8 至 5.5 V				
内部低速振荡时钟（用于 TMH1，WDT）		内部振荡器 240 kHz （典型值）： V_{DD} = 1.8 至 5.5 V				
通用寄存器		8 位 × 32 寄存器 （8 位 × 8 寄存器× 4 组）				
指令最短执行时间		0.2 μs （高速系统时钟： @ f_{XH} = 10 MHz 操作）				
		0.25 μs （内部高速振荡时钟： @ f_{RH} = 8 MHz （典型值） 操作）				
		122 μs （副系统时钟： @ f_{SUB} = 32.768 kHz 操作）				
指令集		• 8 位操作和 16 位操作 • 位操作（置位，重置，测试和布尔操作） • BCD 调整，等				
I/O 端口		总计： 46 CMOS I/O： 42 CMOS 输入： 4				
定时器		• 16 位定时器/事件计数器： 1 通道 • 8 位定时器/事件计数器： 3 通道（其中 2 通道可执行 PWM 输出） • 8 位定时器： 3 通道（其中 2 通道可执行 PWM 输出） • 实时计数器： 1 通道 • 看门狗定时器： 1 通道				
	定时器输出	5 （PWM 输出：4 ，PPG 输出：1）				
	RTC 输出	2 • 1 Hz （副系统时钟： f_{SUB} = 32.768 kHz） • 512 Hz 或 16.384 kHz 或 32.768 kHz （副系统时钟： f_{SUB} = 32.768 kHz）				
蜂鸣器输出		• 1.22 kHz，2.44 kHz，4.88 kHz，9.77 MHz （外设硬件时钟： @ f_{PRS} = 10 MHz 操作）				

注 通过内部存储器容量切换寄存器（IMS）和内部扩展 RAM 容量切换寄存器（IXS），可以改变内部 Flash 存储器、内部高速 RAM 的容量和内部扩展 RAM 的容量。

（2/2）

项目		μPD78F0441	μPD78F0442	μPD78F0443	μPD78F0444	μPD78F0445
10 位逐次逼近型 A/D 转换器		–				
16 位ΔΣ 型 A/D 转换器		–				
串行接口		• 支持 LIN 总线的 UART[注 1]： 1 通道 • 3 线串行 I/O/UART[注 2]： 1 通道				
LCD 控制器/驱动器		• 外部电阻分压和内部电阻分压可切换使用 • Segment 信号输出：32（28） [28（24）] [注 3，4] • 公共端信号输出：4（8）[注 3]				
遥控接收器		提供				
曼彻斯特编码发生器		提供				
中断向量源	内部	19				
	外部	6				
<R> Segment 键源信号输出		Segment 键源信号输出：8（SEG16（KS0）-SEG23（KS7））				
按键中断		通过检测按键输入引脚（KR0 至 KR4）的下降沿，产生按键中断（INTKR）。				
复位		• 使用 $\overline{\text{RESET}}$ 引脚复位 • 看门狗定时器引发的内部复位 • POC 电路引发的内部复位 • 低电压检测电路引发的内部复位				
片上调试功能		提供				
电源电压		V_{DD} = 1.8 至 5.5 V				
工作环境温度		T_A = –40 至 +85°C				
封装形式		• 64 脚塑封 LQFP（密间距）（10 × 10） • 64 脚塑封 LQFP（12 × 12）				

注 **1.** 支持 LIN 总线的 UART 引脚可以更改为 3 线 CSI/UART 引脚（引脚编号 62 和 63）。
2. 在复用功能引脚中可以任意选择一种功能。
3. 括号内的值是使用 8com 时的信号输出数。
<R> **4.** 方括号中的值是使用底部 UART6 引脚（RxD6，TxD6）时的信号输出数量。

定时器概要说明如下

		16 位定时器/事件计数器 00	8 位定时器/事件计数器 50，51，和 52			8 位定时器 H0，H1，和 H2			实时计数器	看门狗定时器
		TM00	TM50	TM51	TM52	TMH0	TMH1	TMH2		
功能	间隔定时器	1 通道	1 通道	1 通道	1 通道	1 通道	1 通道	1 通道	1 通道 注 1	–
	外部事件计数器	1 通道注 2	1 通道	1 通道	1 通道注 2	–	–	–注 2	–	–
	PPG 输出	1 路输出	–	–	–	–	–	–	–	–
	PWM 输出	–	1 路输出	1 路输出	–	1 路输出	1 路输出	–	–	–
	脉冲宽度测量	2 路输入	–	–	–	–	–	–	–	–
	方波输出	1 路输出	1 路输出	1 路输出	–	1 路输出	1 路输出	–	–	–
	载波发生器	–	–	–注 3	–	–	1 路输出 注 3	–	–	–
	日历功能	–	–	–	–	–	–	–	1 通道 注 1	
	RTC 输出	–	–	–	–	–	–	–	2 路输出 注 4	–
	看门狗定时器	–	–	–	–	–	–	–	–	1 通道
中断源		2	1	1	1	1	1	1	1	–

注　**1.** 在实时计数器中，间隔定时器功能和日历功能可以同时使用。

2. TM52 和 TM00 可级联用作 24 位计数器。同样，可以通过 TMH2 来控制 TM52 的外部事件输入使能。

3. TM51 和 TMH1 可联合使用，作为载波发生器模式。

4. 1 Hz 输出可作为 1 路输出，512 Hz、16.384 kHz 或 32.768 kHz 输出可作为 1 路输出。

1.8 功能概述（μPD78F045x）

（1/2）

项目		μPD78F0451	μPD78F0452	μPD78F0453	μPD78F0454	μPD78F0455
内部存储器	Flash 存储器（支持自编程）[注]	16 KB	24 KB	32 KB	48 KB	60 KB
	高速 RAM [注]	768 字节	1 KB			
	扩展 RAM [注]	–			1 KB	
	LCD 显示 RAM	32× 4 位（4 com） 或 28 × 8 位 （8 com）				
存储空间		64 KB				
主系统时钟（振荡频率）	高速系统时钟	主系统时钟 （振荡频率）				
	内部高速振荡时钟					
副系统时钟（振荡频率）		XT1 （晶体） 振荡器 32.768 kHz （典型值）： V_{DD} = 1.8 至 5.5 V				
内部低速振荡时钟（用于 TMH1，WDT）		内部振荡器 240 kHz （典型值）： V_{DD} = 1.8 至 5.5 V				
通用寄存器		8 位 × 32 寄存器 （8 位 × 8 寄存器× 4 组）				
指令最短执行时间		0.2 μs （高速系统时钟： @ f_{XH} = 10 MHz 操作）				
		0.25 μs （内部高速振荡时钟： @ f_{RH} = 8 MHz （典型值） 操作）				
		122 μs （副系统时钟： @ f_{SUB} = 32.768 kHz 操作）				
指令集		• 8 位操作和 16 位操作 • 位操作 （置位，重置，测试和布尔操作） • BCD 调整，等				
I/O 端口		总计：　46 CMOS I/O：　42 CMOS 输入：　4				
定时器		• 16 位定时器/事件计数器：　1 通道 • 8 位定时器/事件计数器：　3 通道（其中 2 通道可执行 PWM 输出） • 8 位定时器：　3 通道（其中 2 通道可执行 PWM 输出） • 实时计数器：　1 通道 • 看门狗定时器：　1 通道				
	定时器输出	5 （PWM 输出： 4 和 PPG 输出： 1）				
	RTC 输出	2 • 1 Hz （副系统时钟： f_{SUB} = 32.768 kHz） • 512 Hz 或 16.384 kHz 或 32.768 kHz （副系统时钟： f_{SUB} = 32.768 kHz）				
蜂鸣器输出		• 1.22 kHz，2.44 kHz，4.88 kHz，9.77 MHz （外设硬件时钟： @ f_{PRS} = 10 MHz 操作）				

注　通过内部存储器容量切换寄存器（IMS）和内部扩展 RAM 容量切换寄存器（IXS），可以改变内部 Flash 存储器、内部高速 RAM 的容量和内部扩展 RAM 的容量。

（2/2）

项目		μPD78F0451	μPD78F0452	μPD78F0453	μPD78F0454	μPD78F0455
10 位逐次逼近型 A/D 转换器		10 位分辨率× 8 通道 （AV_{REF} = 2.3 至 5.5 V）				
16 位ΔΣ 型 A/D 转换器		–				
串行接口		• 支持 LIN 总线的 UART 注 1： 1 通道 • 3 线串行 I/O/UART 注 2： 1 通道				
LCD 控制器/驱动器		• 外部电阻分压和内部电阻分压可切换使用 • Segment 信号输出： 32（28） [28（24）] 注 3，4 • 公共端信号输出： 4 （8） 注 3				
遥控接收器		提供				
曼彻斯特编码发生器		提供				
中断向量源	内部	20				
	外部	6				
<R> Segment 键源信号输出		Segment 键源信号输出：8（SEG16（KS0）-SEG23（KS7））				
按键中断		通过检测按键输入引脚（KR0 至 KR4）的下降沿，产生按键中断（INTKR）。				
复位		• 使用 $\overline{\text{RESET}}$ 引脚复位 • 看门狗定时器引发的内部复位 • POC 电路引发的内部复位 • 低电压检测电路引发的内部复位				
片上调试功能		提供				
电源电压		V_{DD} = 1.8 至 5.5 V				
工作环境温度		T_A = –40 至 +85°C				
封装形式		• 64 脚塑封 LQFP（密间距）（10 × 10） • 64 脚塑封 LQFP（12 × 12）				

注 **1.** 支持 LIN 总线的 UART 引脚可以更改为 3 线 CSI 引脚（引脚编号 62 和 63）。
2. 在复用功能引脚中可以任意选择一种功能。
3. 括号内的值是使用 8com 时的信号输出数。
<R> **4.** 方括号中的值是使用底部 UART6 引脚（RxD6，TxD6）时的信号输出数量。

定时器的概况如下

		16 位定时器/事件计数器 00	8 位定时器/事件计数器 50，51，和 52			8 位定时器 H0，H1，和 H2			实时计数器	看门狗定时器
		TM00	TM50	TM51	TM52	TMH0	TMH1	TMH2		
功能	间隔定时器	1 通道	1 通道	1 通道	1 通道	1 通道	1 通道	1 通道	1 通道[注 1]	–
	外部事件计数器	1 通道[注 2]	1 通道	1 通道	1 通道[注 2]	–	–	–[注 2]	–	–
	PPG 输出	1 路输出	–	–	–	–	–	–	–	–
	PWM 输出	–	1 路输出	1 路输出	–	1 路输出	1 路输出	–	–	–
	脉冲宽度测量	2 路输入	–	–	–	–	–	–	–	–
	方波输出	1 路输出	1 路输出	1 路输出	–	1 路输出	1 路输出	–	–	–
	载波发生器	–	–	–[注 3]	–	–	1 路输出[注 3]	–	–	–
	日历功能	–	–	–	–	–	–	–	1 通道[注 1]	
	RTC 输出	–	–	–	–	–	–	–	2 路输出[注 4]	–
	看门狗定时器	–	–	–	–	–	–	–	–	1 通道
中断源		2	1	1	1	1	1	1	1	–

注 **1.** 在实时计数器中，间隔定时器功能和日历功能可以同时使用。

2. TM52 和 TM00 可级联用作 24 位计数器。同样，可以通过 TMH2 来控制 TM52 的外部事件输入使能。

3. TM51 和 TMH1 可联合使用，作为载波发生器模式。

4. 1 Hz 输出可作为 1 路输出，512 Hz、16.384 kHz 或 32.768 kHz 输出可作为 1 路输出。

1.9 功能概述（μPD78F046x）

项目		μPD78F0461	μPD78F0462	μPD78F0463	μPD78F0464	μPD78F0465
内部存储器	Flash 存储器（支持自编程）[注]	16 KB	24 KB	32 KB	48 KB	60 KB
	高速 RAM [注]	768 字节	1 KB			
	扩展 RAM [注]	–			1 KB	
	LCD 显示 RAM	24 × 4 位（4 com） 或 20 × 8 位 （8 com）				
存储空间		64 KB				
主系统时钟（振荡频率）	高速系统时钟	X1 （晶体/陶瓷） 振荡器， 外部主系统时钟输入（EXCLK） 2 至 10 MHz： V_{DD} = 2.7 至 5.5 V， 2 至 5 MHz： V_{DD} = 1.8 至 5.5 V				
	内部高速振荡时钟	内部振荡器 8 MHz （典型值）： V_{DD} = 1.8 至 5.5 V				
副系统时钟（振荡频率）		XT1 （晶体） 振荡器 32.768 kHz （典型值）： V_{DD} = 1.8 至 5.5 V				
内部低速振荡时钟（用于 TMH1，WDT）		内部振荡器 240 kHz （典型值）： V_{DD} = 1.8 至 5.5 V				
通用寄存器		8 位 × 32 寄存器 （8 位 × 8 寄存器× 4 组）				
指令最短执行时间		0.2 μs （高速系统时钟： @ f_{XH} = 10 MHz 操作）				
		0.25 μs （内部高速振荡时钟： @ f_{RH} = 8 MHz （典型值） 操作）				
		122 μs （副系统时钟： @ f_{SUB} = 32.768 kHz 操作）				
指令集		• 8 位操作和 16 位操作 • 位操作 （置位，重置，测试和布尔操作） • BCD 调整，等				
I/O 端口		总计： 46 CMOS I/O： 42 CMOS 输入： 4				
定时器		• 16 位定时器/事件计数器： 1 通道 • 8 位定时器/事件计数器： 3 通道（其中 2 通道可执行 PWM 输出） • 8 位定时器： 3 通道（其中 2 通道可执行 PWM 输出） • 实时计数器： 1 通道 • 看门狗定时器： 1 通道				
	定时器输出	5 （PWM 输出： 4 ，PPG 输出： 1）				
	RTC 输出	2 • 1 Hz （副系统时钟： f_{SUB} = 32.768 kHz） • 512 Hz 或 16.384 kHz 或 32.768 kHz （副系统时钟： f_{SUB} = 32.768 kHz）				
蜂鸣器输出		• 1.22 kHz，2.44 kHz，4.88 kHz，9.77 MHz （外设硬件时钟： @ f_{PRS} = 10 MHz 操作）				

注　通过内部存储器容量切换寄存器（IMS）和内部扩展 RAM 容量切换寄存器（IXS），可以改变内部 Flash 存储器、内部高速 RAM 的容量和内部扩展 RAM 的容量。

（2/2）

项目		μPD78F0461	μPD78F0462	μPD78F0463	μPD78F0464	μPD78F0465
10 位逐次逼近型 A/D 转换器		10 位分辨率× 8 通道 （AV_{REF} = 2.3 至 5.5 V）				
16 位ΔΣ 型[注 1] A/D 转换器		16 位分辨率× 3 通道（AV_{REF} = 2.7 至 5.5 V）				
串行接口		• 支持 LIN 总线的 UART[注 2]： 1 通道 • 3 线串行 I/O/UART[注 3]： 1 通道				
LCD 控制器/驱动器		• 外部电阻分压和内部电阻分压可切换使用 • Segment 信号输出： 24（20） [20（16）][注 4，5] • 公共端 信号输出： 4 （8）[注 4]				
遥控接收器		提供				
曼彻斯特编码发生器		提供				
中断向量源	内部	21				
	外部	6				
<R> Segment 键源信号输出		Segment 键源信号输出：8（SEG16（KS0）-SEG23（KS7））				
按键中断		通过检测按键输入引脚（KR0 至 KR4）的下降沿，产生按键中断（INTKR）。				
复位		• 使用 $\overline{\text{RESET}}$ 引脚复位 • 看门狗定时器引发的内部复位 • POC 电路引发的内部复位 • 低电压检测电路引发的内部复位				
片上调试功能		提供				
电源电压		V_{DD} = 1.8 至 5.5 V				
工作环境温度		T_A = –40 至 +85°C				
封装形式		• 64 脚塑封 LQFP（密间距）（10 × 10） • 64 脚塑封 LQFP（12 × 12）				

注 **1.** 16 位 A/D 转换器的规格可能已经更改。
规格的详情，请联系 NEC 电子销售代表或授权的经销商。
2. 支持 LIN 总线的 UART 引脚可以更改为 3 线 CSI/引脚（引脚编号 62 和 63）。
3. 在复用功能引脚中可以任意选择一种功能。
4. 括号内的值是使用 8com 时的信号输出数。
<R> **5.** 方括号中的值是使用底部 UART6 引脚（RxD6，TxD6）时的信号输出数量。

定时器概要说明如下

		16 位定时器/事件计数器 00	8 位定时器/事件计数器 50，51，和 52			8 位定时器 H0，H1，和 H2			实时计数器	看门狗定时器
		TM00	TM50	TM51	TM52	TMH0	TMH1	TMH2		
功能	间隔定时器	1 通道	1 通道	1 通道	1 通道	1 通道	1 通道	1 通道	1 通道 注 1	–
	外部事件计数器	1 通道注 2	1 通道	1 通道	1 通道注 2	–	–	–注 2	–	–
	PPG 输出	1 路输出	–	–	–	–	–	–	–	–
	PWM 输出	–	1 路输出	1 路输出	–	1 路输出	1 路输出	–	–	–
	脉冲宽度测量	2 路输入	–	–	–	–	–	–	–	–
	方波输出	1 路输出	1 路输出	1 路输出	–	1 路输出	1 路输出	–	–	–
	载波发生器	–	–	–注 3	–	–	1 路输出 注 3	–	–	–
	日历功能	–	–	–	–	–	–	–	1 通道 注 1	
	RTC 输出	–	–	–	–	–	–	–	2 路输出注 4	–
	看门狗定时器	–	–	–	–	–	–	–	–	1 通道
中断源		2	1	1	1	1	1	1	1	–

注 **1.** 在实时计数器中，间隔定时器功能和日历功能可以同时使用。

2. TM52 和 TM00 可级联用作 24 位计数器。同样，可以通过 TMH2 来控制 TM52 的外部事件输入使能。

3. TM51 和 TMH1 可联合使用，作为载波发生器模式。

4. 1 Hz 输出可作为 1 路输出，512 Hz、16.384 kHz 或 32.768 kHz 输出可作为 1 路输出。

第二章 引脚功能

2.1 引脚功能列表

有三种类型的引脚 输入/输出 缓冲器供电电源：AV_{REF}[注 1]，V_{LC0} 和 V_{DD}。下表显示了这些供电电源与引脚之间的关系。

表 2-1. 引脚 输入/输出缓冲器供电电源

供电电源	对应引脚
AV_{REF}[注 1]	P20 至 P27
V_{LC0}	COM0 至 COM7，SEG0 至 SEG23，SEG24 至 SEG31[注 2]， V_{LC0} 至 V_{LC3}
V_{DD}	其它引脚

注 **1.** 仅限μPD78F045x 和 78F046x。μPD78F044x 的供电电源是 V_{DD}。
2. 仅限μPD78F044x 和 78F045x 。

（1） 端口引脚 **（1/2）**

引脚名称	输入/输出	功能	复位后	复用功能
P11	输入/输出	端口 1 4 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。	输入端口	$\overline{SCK10}$
P12				SI10/RxD0/<RxD6>
P13				SO10/TxD0/<TxD6>
P14				INTP4
P20	输入/输出	端口 2 8 位 输入/输出 端口。 可以按位选择输入/输出模式。	数字输入端口	SEG31[注 1]/ANI0[注 2]/DS0–[注 3]
P21				SEG30[注 1]/ANI1[注 2]/DS0+[注 3]
P22				SEG29[注 1]/ANI2[注 2]/DS1–[注 3]
P23				SEG28[注 1]/ANI3[注 2]/DS1+[注 3]
P24				SEG27[注 1]/ANI4[注 2]/DS2–[注 3]
P25				SEG26[注 1]/ANI5[注 2]/DS2+[注 3]
P26				SEG25[注 1]/ANI6[注 2]/REF–[注 3]
P27				SEG24[注 1]/ANI7[注 2]/REF+[注 3]

注 **1.** 仅限μPD78F044x 和 78F045x 。
2. 仅限μPD78F045x 和 78F046x 。
3. 仅限μPD78F046x。

备注 尖括号（< >） 内的功能由输入转换控制寄存器（ISC）分配。

（1） 端口引脚 （2/2）

引脚名称	输入/输出	功能	复位后	复用功能
P31	输入/输出	端口 3 4 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。	输入端口	TOH1/INTP3
P32				TOH0/MCGO
P33				TI000/RTCDIV/ RTCCL/BUZ/INTP2
P34				TI52/TI010/TO00/ RTC1HZ/INTP1
P40	输入/输出	端口 4 5 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。	输入端口	V_{LC3}/KR0
P41				RIN/KR1
P42				KR2
P43				TO51/TI51/KR3
P44				TO50/TI50/KR4
P80 至 P83	输入/输出	端口 8 4 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。	输入端口	SEG4 至 SEG7
P100 至 P103	输入/输出	端口 10 4 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。	输入端口	SEG8 至 SEG11
P110，P111	输入/输出	端口 11 4 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。	输入端口	SEG12, SEG13
P112				SEG14/TxD6
P113				SEG15/RxD6
P120	输入/输出	端口 12. 1 位输入/输出端口和 4 位 输入端口。 可以按位选择输入/输出模式。 仅有 P120 可以通过软件设置指定内置上拉电阻的使用。	输入端口	INTP0/EXLVI
P121	输入			X1/OCD0A
P122				X2/EXCLK/OCD0B
P123				XT1
P124				XT2
P140 至 P143	输入/输出	端口 14 4 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。.	输入端口	SEG16 (KS0) 至 SEG19 (KS3)
P150 至 P153	输入/输出	端口 15 4 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。	输入端口	SEG20 (KS4) 至 SEG23 (KS7)

（2） 非端口引脚　　　　（1/3）

功能名称	输入/输出	功能	复位后	复用功能
ANI0 [注2]	输入	10 位逐次逼近型 A/D 转换器模拟输入	数字输入端口	P20/SEG31 [注1]/ DS0– [注3]
ANI1 [注2]				P21/SEG30 [注1]/ DS0+ [注3]
ANI2 [注2]				P22/SEG29 [注1]/ DS1– [注3]
ANI3 [注2]				P23/SEG28 [注1]/ DS1+ [注3]
ANI4 [注2]				P24/SEG27 [注1]/ DS2– [注3]
ANI5 [注2]				P25/SEG26 [注1]/ DS2+ [注3]
ANI6 [注2]				P26/SEG25 [注1]/ REF– [注3]
ANI7 [注2]				P27/SEG24 [注1]/ REF+ [注3]
DS0– [注3]	输入	16 位 ΔΣ 型 A/D 转换器模拟输入	数字输入端口	P20/ANI0 [注2]
DS0+ [注3]				P21/ANI1 [注2]
DS1– [注3]				P22/ANI2 [注2]
DS1+ [注3]				P23/ANI3 [注2]
DS2– [注3]				P24/ANI4 [注2]
DS2+ [注3]				P25/ANI5 [注2]
REF– [注3]		16 位 ΔΣ 型 A/D 转换器参考电压输入。 应该使 V_{SS} 和 AV_{SS} 具有相同电平。		P26/ANI6 [注2]
REF+ [注3]		16 位 ΔΣ 型 A/D 转换器参考电压输入。 应该和 AV_{REF} 具有相同电平。		P27/ANI7 [注2]
AV_{REF} [注2]	输入	10 位逐次逼近型 A/D 转换器参考电压输入， 端口 2 和 16 位 ΔΣ 型 A/D 转换器[注3]的正向电源。	–	–
AV_{SS} [注2]	–	A/D 转换地电位。应该和 V_{SS} 具有相同电平	–	–
SEG0 至 SEG3	输出	LCD 控制器/驱动器 segment 信号输出	输出	COM4 至 COM7
SEG4 至 SEG7			输入端口	P80 至 P83
SEG8 至 SEG11				P100 至 P103
SEG12，SEG13				P110, P111
SEG14				P112/TxD6
SEG15				P113/RxD6
<R> SEG16 (KS0) 至 SEG19 (KS3)		LCD 控制器/驱动器 segment 信号输出。 可以同时输出 Segment 键源信号。		P140 至 P143
<R> SEG20 (KS4) 至 SEG23 (KS7)				P150 至 P153

注　**1.** 仅限μPD78F044x 和 78F045x 。
　　2. 仅限μPD78F045x 和 78F046x 。
　　3. 仅限μPD78F046x 。

（2） 非端口引脚 （2/3）

功能名称	输入/输出	功能	复位后	复用功能
SEG24[注 1]		LCD 控制器/驱动器 segment 信号输出	数字输入端口	P27/ANI7[注 2]
SEG25[注 1]				P26/ANI6[注 2]
SEG26[注 1]				P25/ANI5[注 2]
SEG27[注 1]				P24/ANI4[注 2]
SEG28[注 1]				P23/ANI3[注 2]
SEG29[注 1]				P22/ANI2[注 2]
SEG30[注 1]				P21/ANI1[注 2]
SEG31[注 1]				P20/ANI0[注 2]
COM0 至 COM3	输出	LCD 控制器/驱动器 segment 信号输出	输出	–
COM4 至 COM7				SEG0 至 SEG3
V_{LC0} 至 V_{LC2}	–	LCD 驱动电压	–	–
V_{LC3}			输入端口	P40/KR0
BUZ	输出	蜂鸣器输出	输入端口	P33/TI000/RTCDIV /RTCCL/INTP2
INTP0	输入	定义外部中断请求输入的有效边沿（上升沿、下降沿，或上升和下降双沿）。	输入端口	P120/EXLVI
INTP1				P34/TI52/TI010/ TO00/RTC1HZ
INTP2				P33/TI000/RTCDIV /RTCCL/BUZ
INTP3				P31/TOH1
INTP4				P14
<R> KR0	输入	按键中断输入	输入端口	P40/V_{LC3}
KR1				P41/RIN
KR2				P42
KR3				P43/TO51/TI51
KR4				P44/TO50/TI50
MCGO	输出	曼彻斯特编码输出	输入端口	P32/TOH0
REGC	–	为了内部操作，请连接稳压器输出（2.4V） 稳定电容。 通过一个电容器（0.47 μF 至 1μF：推荐） 连接至 V_{SS}。	–	–
$\overline{\text{RESET}}$	输入	系统复位输入	–	–
RIN	输入	遥控接收数据输入	输入端口	P41/KR1
RTCDIV	输出	实时计数器时钟（32 kHz 分频） 输出	输入端口	P33/TI000/RTCCL /BUZ/INTP2
RTCCL	输出	实时计数器时钟（32 kHz 振荡） 输出	输入端口	P33/TI000/RTCDIV /BUZ/INTP2
RTC1HZ	输出	实时计数器时钟（1 kHz ） 输出	输入端口	P34/TI52/TI010/ TO00/INTP1

注 **1.** 仅限μPD78F044x 和 78F045x。
2. 仅限μPD78F045x 和 78F046x 。

（2） 非端口引脚 （3/3）

功能名称	输入/输出	功能	复位后	复用功能
RxD0	输入	异步串行接口的串行数据输入	输入端口	P12/SI10/<RxD6>
RxD6				P113/SEG15
<RxD6>				P12/SI10/RxD0
SI10	输入	CSI10 串行数据输入	输入端口	P12/RxD0/<RxD6>
SO10	输出	CSI10 串行数据输出	输入端口	P13/TxD0/<TxD6>
$\overline{SCK10}$	输入/输出	串行接口 CSI10 的时钟输入/输出	输入端口	P11
TI000	输入	16 位定时器/事件计数器 00 的外部计数时钟输入 16 位定时器/事件计数器 00 捕捉寄存器（CR000、CR010）的捕捉触发输入	输入端口	P33/RTCDIV/RTCCL/BUZ/INTP2
TI010		16 位定时器/事件计数器 00 捕捉寄存器（CR000）的捕捉触发输入		P34/TI52/TO00/RTC1HZ/INTP1
TI50	输入	8 位定时器/事件计数器 50 的外部计数时钟输入	输入端口	P44/TO50/KR4
TI51		8 位定时器/事件计数器 51 的外部计数时钟输入		P43/TO51/KR3
TI52		8 位定时器/事件计数器 52 的外部计数时钟输入		P34/TI010/TO00/RTC1HZ/INTP1
TO00	输出	16 位定时器/事件计数器 00 输出	输入端口	P34/TI52/TI010/RTC1HZ/INTP1
TO50	输出	8 位定时器/事件计数器 50 输出	输入端口	P44/TI50/KR4
TO51		8 位定时器/事件计数器 51 输出		P43/TI51/KR3
TOH0	输出	8 位定时器 H0 输出	输入端口	P32/MCGO
TOH1		8 位定时器 H1 输出		P31/INTP3
TxD0	输出	异步串行接口的串行数据输出	输入端口	P13/SO10/<TxD6>
TxD6				P112/SEG14
<TxD6>				P13/SO10/TxD0
EXLVI	输入	外部低电压检测的电位输入	输入端口	P120/INTP0
X1	输入	为主系统时钟连接谐振器	输入端口	P121/OCD0A
X2	–			P122/EXCLK/OCD0B
EXCLK	输入	用于外部低电压检测的电压输入	输入端口	P122/X2/OCD0B
XT1	输入	为副系统时钟连接谐振器	输入端口	P123
XT2	–			P124
V_{DD}	–	正向电源	–	–
V_{SS}	–	地	–	–
FLMD0	–	Flash 存储器编程模式设置	–	–
OCD0A	输入	用于设置片上调试模式	输入端口	P121/X1
OCD0B	–			P122/X2/EXCLK

备注 尖括号（< >） 内的功能由输入转换控制寄存器（ISC）分配。

2.2 引脚功能描述

2.2.1 P11 至P14（端口 1）

P11 至 P14 用作 4 位 输入/输出端口。这些引脚也可用于外部中断请求输入，串行输入输入/输出和时钟输入/输出。P13 可由端口功能寄存器（PF1）选择作为引脚功能（参见图 4-22）。

以下操作模式可以按位指定。

（1） 端口模式

P11 至 P14 作为 4 位 输入/输出端口使用。通过使用端口模式寄存器 1（PM1），可按位设置 P11 至 P14 为输入输出端口。由上拉电阻选择寄存器 1（PU1）定义内置上拉电阻的使用。

（2） 控制模式

P11 至 P14 也可用于外部中断请求输入，串行输入输入/输出和时钟输入/输出。

（a） SI10

串行接口的串行数据输入引脚。

（b） SO10

串行接口的串行数据输出引脚。

（c） $\overline{\text{SCK10}}$

串行接口的串行时钟输入/输出引脚。

（d） RxD0， RxD6

异步串行接口的串行数据输入引脚。

（e） TxD0， TxD6

异步串行接口的串行数据输出引脚。

（f） INTP4

是外部中断请求输入引脚，可以指定有效沿（上升沿、下降沿，或上升和下降双沿）。

2.2.2 P20 至P27（端口 2）

P20 至 P27 用作 8 位输入/输出端口。这些引脚也可用于 LCD 控制器/驱动器的 segment 信号输出、10 位次逼近型 A/D 转换器模拟输入（仅限μPD78F045x 和 78F046x）、16 位 ΔΣ 型 A/D 转换器模拟输入和参考电压输入（仅限μPD78F046x）。可由端口功能寄存器（PF2）选择作为输入/输出引脚功能或 segment 信号输出功能。

以下操作模式可以按位指定。

（1） 端口模式

P20 至 P27 作为 8 位 输入/输出端口使用。通过使用端口模式寄存器 2（PM2），可按位设置 P20 至 P27 为输入输出端口。

（2） 控制模式

P20 至 P27 可用于 LCD 控制器/驱动器的 segment 信号输出、10 位次逼近型 A/D 转换器模拟输入（仅限μPD78F045x 和 78F046x）、16 位 ΔΣ 型 A/D 转换器模拟输入和参考电压输入（仅限μPD78F046x）。

（a） SEG24 至 SEG31

这些引脚可用于 LCD 控制器/驱动器的 segment 信号输出。

（b） ANI0 至 ANI7（仅限μPD78F045x 和 78F046x ）

这些是 10 位次逼近型 A/D 转换器模拟输入引脚。当使用这些引脚作为模拟输入引脚时，参见 **12.6 10 位次逼近型 A/D 转换器注意事项**中的（**5**）**ANI0/SEG31/P20 至 ANI7/SEG24/P27 引脚（ PD78F045x），ANI0/DS0–/P20 至 ANI7/REF+/P27 引脚（ PD78F046x）**。

（c）DS0–，DS0+，DS1–，DS1+，DS2–，DS2+，REF–和 REF+（仅限μPD78F046x ）

这些是 16 位 ΔΣ 型 A/D 转换器模拟输入和参考电压输入引脚。

设置 REF– 与 V_{SS} 和 AV_{SS} 具有相同的电平。

设置 REF+ 与 AV_{REF} 具有相同的电平。

注意事项 **复位释放后，P20 至 P27 被设置为模拟输入模式。**

2.2.3 P31 至P34（端口 3）

P31 至 P34 用作 4 位 输入/输出端口。这些引脚也可用于外部中断请求输入、定时器输入/输出、蜂鸣器输出、实时计数器输出和曼彻斯特编码输出。

以下操作模式可以按位指定。

（1） 端口模式

P31 至 P34 用作 4 位输入/输出端口。通过使用端口模式寄存器 3（PM3），可按位设置 P31 至 P34 为输入输出端口。由上拉电阻选择寄存器 3（PU3）定义内置上拉电阻的使用。

（2） 控制模式

P31 至 P34 也可用作外部中断请求输入、定时器输入/输出、蜂鸣器输出、实时计数器输出和曼彻斯特编码输出。

（a） INTP1 至 INTP3

这些是外部中断请求输入引脚，可指定有效边沿（上升沿、下降沿，或上升和下降双沿）。

（b） TO00，TOH0，TOH1

定时器输出引脚。

（c） TI000

通过该引脚将外部计数时钟输入到 16 位定时器/事件计数器 00，也可将捕捉触发信号输入到 16 位定时器/事件计数器 00 的捕捉寄存器（CR000 或 CR010）中。

（d） TI010

将捕捉触发信号输入到 16 位定时器/事件计数器 00 的捕捉寄存器（CR000）中。

（e） TI52

将外部计数时钟输入到 8 位定时器/事件计数器 52。

（f） BUZ

蜂鸣器输出引脚。

（g） RTCDIV

实时计数器时钟（32 kHz 分频） 输出引脚。

（h） RTCCL

实时计数器时钟（32 kHz 振荡） 输出引脚。

（i） RTC1HZ

实时计数器修正时钟（1 kHz 振荡） 输出引脚。

（j） MCGO

曼彻斯特编码输出引脚。

2.2.4 P40 至P44（端口 4）

P40 至 P44 用作 5 位 输入/输出端口。这些引脚也可用于按键中断输入、segment 键扫描输入、定时器输入/输出、遥控接收数据输入及 LCD 驱动电源。

以下操作模式可以按位指定。

（1） 端口模式

P40 至 P44 用作 5 位 输入/输出端口使用。通过使用端口模式寄存器 4（PM4），可按位设置 P40 至 P44 为输入输出端口。由上拉电阻选择寄存器 4（PU4）定义内置上拉电阻的使用。

（2） 控制模式

P40 至 P44 可用于按键中断输入、segment 键扫描输入、定时器输入/输出、遥控接收数据输入及 LCD 驱动电源。

（a） KR0 至 KR4

<R> 这些是按键中断输入或 segment 键扫描输入引脚。

（b） TO50，TO51

8 位定时器/事件计数器 50 和 51 的定时器输出引脚。

（c） TI50，TI51

用于将外部计数时钟输入到 8 位定时器/事件计数器 50 和 51。

（d） RIN

遥控接收器的数据输入。

（e） V_{LC3}

驱动 LCD 的供电电压引脚。

2.2.5 P80 至P83（端口 8）

P80 至 P83 用作 4 位输入/输出端口。这些引脚也可以作为 LCD 控制器/驱动器的 segment 信号输出。由端口功能寄存器 ALL（PFALL）选择作为输入/输出 端口或 segment 信号输出。

.

（1） 端口模式

P80 至 P83 用作 4 位输入/输出端口。通过使用端口模式寄存器 8（PM8），可按位设置 P80 至 P83 为输入输出端口。由上拉电阻选择寄存器 8（PU8）定义内置上拉电阻的使用。

（2） 控制模式

P80 至 P83 可作为 LCD 控制器/驱动器的 segment 信号输出。

（a） SEG4 至 SEG7

这些是 LCD 控制器/驱动器的 segment 信号输出引脚。

2.2.6 P100 至P103（端口 10）

P100 至 P103 用作 4 位输入/输出端口。这些引脚也可以作为 LCD 控制器/驱动器的 segment 信号输出。由端口功能寄存器 ALL（PFALL）选择作为输入/输出 端口功能或 segment 信号输出功能。

（1） 端口模式

P100 至 P103 用作 4 位输入/输出端口。通过使用端口模式寄存器 10（PM10），可按位设置 P100 至 P103 为输入输出端口。由上拉电阻选择寄存器 10（PU10）定义内置上拉电阻的使用。

（2） 控制模式

P100 至 P103 可作为 LCD 控制器/驱动器的 segment 信号输出。

（a） SEG8 至 SEG11

这些是 LCD 控制器/驱动器的 segment 信号输出引脚。

2.2.7 P110 至P113（端口 11）

P110 至 P113 用作 4 位输入/输出端口。这些引脚也可以作为 LCD 控制器/驱动器的 segment 信号输出和串行接口数据输入/输出。由端口功能寄存器 ALL（PFALL）选择作为输入/输出 端口功能（除了 segment 信号输出）或 segment 信号输出功能。

（1） 端口模式

P110 至 P113 用作 4 位输入/输出端口。通过使用端口模式寄存器 11（PM11），可按位设置 P110 至 P113 为输入输出端口。由上拉电阻选择寄存器 11（PU11）定义内置上拉电阻的使用。

（2） 控制模式

P110 至 P113 可作为 LCD 控制器/驱动器的 segment 信号输出和串行接口数据输入/输出。

（a） SEG12 至 SEG15

这些是 LCD 控制器/驱动器的 segment 信号输出引脚。

（b） RxD6

串行接口 UART6 的串行数据输入引脚。

（c） TxD6

串行接口 UART6 的串行数据输出引脚。

2.2.8 P120 至P124（端口 12）

P120 用作 1 位输入/输出端口。P121 至 P124 用作 4 位输入端口。这些引脚也可以作为外部中断请求输入、外部低电压检测的电压输入、连接主系统时钟振荡器、连接副系统时钟振荡器和外部时钟输入。以下操作模式可以按位指定。

（1） 端口模式

P120 用作 1 位输入/输出端口，且 P121 至 P124 用作 4 位输入端口。仅有 P120 可由端口模式寄存器 12（PM12）设置为输入或输出端口。仅有 P120 可由上拉电阻选择寄存器 12（PU12）定义内置上拉电阻的使用。

（2） 控制模式

P120 至 P124 可作为外部中断请求输入、外部低电压检测的电压输入、连接主系统时钟的振荡器、连接副系统时钟的振荡器和外部时钟输入。

（a） INTP0

作为外部中断请求输入（INTP0）使用，可以指定有效边沿（上升沿、下降沿，或上升和下降双沿）。

（b） EXLVI

用于外部低电压检测的电位输入。

（c）X1，X2

用于连接主系统时钟振荡器。

（d） EXCLK

用于主系统时钟的外部时钟输入。

（e） XT1，XT2

用于连接副系统时钟振荡器。

备注　使用片上调试功能时，X1 和 X2 可以被用作片上调试模式设置引脚（OCD0A，OCD0B）。详情参见 **第二十八章 片上调试功能**。

2.2.9 P140 至P143（端口 14）

P140 至 P143 用作 4 位输入/输出端口。这些引脚也可以作为 LCD 控制器/驱动器的 segment 信号输出和 segment 键源信号的同时输出。由端口功能寄存器 ALL（PFALL）选择作为输入/输出 端口功能或 segment 信号输出功能。

（1） 端口模式

P140 至 P143 用作 4 位输入/输出端口。通过使用端口模式寄存器 14（PM14），可按位设置 P140 至 P143 为输入输出端口。由上拉电阻选择寄存器 14（PU14）定义内置上拉电阻的使用。

（2） 控制模式

P140 至 P143 可作为 LCD 控制器/驱动器的 segment 信号输出和 segment 键源信号的同时输出。

<R> **（a） SEG16（KS0）至 SEG19（KS3）**

这些是 LCD 控制器/驱动器的 segment 信号输出引脚。

通过设置 LCD 模式寄存器（LCDMD）可以同时输出 segment 键源信号。

2.2.10 P150 至P153（端口 15）

P150 至 P153 用作 4 位输入/输出端口。这些引脚也可以作为 LCD 控制器/驱动器的 segment 信号输出和 segment 键源信号的同时输出。由端口功能寄存器 ALL（PFALL）选择作为输入/输出 端口功能或 segment 信号输出功能。

（1） 端口模式

P150 至 P153 用作 4 位输入/输出端口。通过使用端口模式寄存器 15（PM15），可按位设置 P150 至 P153 为输入输出端口。由上拉电阻选择寄存器 15（PU15）定义内置上拉电阻的使用。

（2） 控制模式

P150 至 P153 可作为 LCD 控制器/驱动器的 segment 信号输出和 segment 键源信号的同时输出。

<R> **（a） SEG20（KS4） 至 SEG23（KS7）**

这些是 LCD 控制器/驱动器的 segment 信号输出引脚。

通过设置 LCD 模式寄存器（LCDMD）可以同时输出 segment 键源信号。

2.2.11 AV_{REF}（仅限μPD78F045x 和 78F046x）

10 位逐次逼近型 A/D 转换器参考电压输入引脚，端口 2 和 16 位 ΔΣ 型 A/D 转换器正向电源。
不使用 A/D 转换器时，将该引脚直接连到 V_{DD} 注

注　　当端口 2 的一个或更多引脚作为数字端口使用时，应该使 AV_{REF} 引脚电平与 V_{DD} 引脚的电平相同。

2.2.12 AV_{SS}（仅限μPD78F045x 和 78F046x）

作为 A/D 转换器的地电位引脚。即使在不使用 A/D 转换器时，该引脚电平也始终应与 V_{SS} 相同

2.2.13 COM0 至COM7

这些引脚作为 LCD 控制器/驱动器的的公共端信号输出引脚。

2.2.14 V_{LC0} 至V_{LC3}

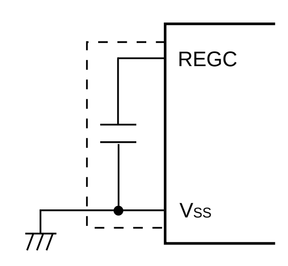
这些引脚作为驱动 LCD 的电源电压引脚。

2.2.15 $\overline{\text{RESET}}$

低有效的系统复位输入引脚。

2.2.16 REGC

为了内部操作，该引脚用于连接调节器输出（2.4 V）稳定电容。通过一个电容（0.47 至 1μF：推荐值）将此引脚连接到 V_{SS} 。



注意事项　　**在上图虚线内的部分，请使用尽可能短的线连接。**

2.2.17 V_{DD}

这是正向供电电源引脚。

2.2.18 V_{SS}

这是地电位引脚。

2.2.19 FLMD0

用于 Flash 存储器编程模式设置。
在正常操作模式下将 FLMD0 连接至 V_{SS}。
在 Flash 存储器编程模式下，将该引脚连接至 Flash 编程器。

2.3 引脚 输入/输出电路和未使用引脚的建议连接方式

表 2-2 为引脚输入/输出电路类型和未使用引脚的建议连接方式。
关于每种类型的输入/输出电路结构，参见图 **2-1**。

表 2-2. 引脚输入/输出电路类型（1/2）

引脚名称	输入/输出 电路类型	输入/输出	未使用引脚的建议连接方式
P11/$\overline{SCK10}$	5-AH	输入/输出	输入：　通过电阻单独连接至 V_{DD} 或 V_{SS} 。 输出：　保持开路。
P12/SI10/RxD0/<RxD6>			
P13/SO10/TxD0/<TxD6>			
P14/INTP4			
P20/SEG31/ANI0/DS0- 至 P27/SEG24/ANI7/REF+ 注s 1, 2, 3, 4	17-R		<模拟设置> 连接至 AV_{REF} 或 AV_{SS}。 <数字设置> 输入：　通过电阻单独连接至 V_{DD} 或 V_{SS} [注 5]。 输出：　保持开路。 <Segment 设置> 保持开路。
P31/TOH1/INTP3	5-AH		输入：　通过电阻单独连接至 V_{DD} 或 V_{SS} 。 输出：　保持开路。
P32/TOH0/MCGO	5-AG		
P33/TI000/RTCDIV/ RTCCL/BUZ/INTP2	5-AH		
P34/TI52/TI010/TO00/ RTC1HZ/INTP1			
P40/V_{LC3}/KR0	5-AO		
P41/RIN/KR1	5-AH		
P42/KR2			
P43/TO51/TI51/KR3			
P44/TO50/TI50/KR4			
P80/SEG4 至 P83/SEG7	17-P		<端口设置> 输入：　通过电阻单独连接至 V_{DD} 或 V_{SS} 。 输出：　保持开路。 <Segment 设置> 保持开路。
P100/SEG8 至 P103/SEG11			
P110/SEG16, P111/SEG17			
P112/SEG18/TxD6			
P113/SEG19/RxD6	17-Q		

注　**1.** 仅对μPD78F044x 和 78F045x 提供 SEGx。
　　2. 仅对μPD78F045x 和 78F046x 提供 ANIx。
　　3. 仅对 78F046x 提供 DSx 和 REFx。
　　4. 复位释放后，P20/SEG31/ANI0/DS0- 至 P27/SEG24/ANI7/REF+被设置为数字输入模式。
　　5. 在使用　PD78F044x 时，通过电阻单独连接至 V_{DD} 或 V_{SS}。

备注　箭头（< >）内的功能由输入转换控制寄存器（ISC）分配。

表 **2-2.** 引脚输入**/**输出电路类型（**2/2**）

引脚名称	输入/输出 电路类型	输入/输出	未使用引脚的建议连接方式
P120/INTP0/EXLVI	5-AH	输入/输出	输入： 通过电阻单独连接至 V_{DD} 或 V_{SS} 。 输出： 保持开路。
P121/X1/OCD0A [注1]	37-A	输入	通过电阻单独连接至 V_{DD} 或 V_{SS} 。
P122/X2/EXCLK/OCD0B [注1]			
P123/XT1 [注1]			
P124/XT2 [注1]			
P140/SEG16 (KS0) 至 P143/SEG19 (KS3)	17-P	输入/输出	<端口设置> 输入： 通过电阻单独连接至 V_{DD} 或 V_{SS} 。 输出： 保持开路。 <Segment 设置> 保持开路。
P150/SEG20 (KS4) 至 P153/SEG23 (KS7)			
COM0 至 COM3	18-E	输出	保持开路。
COM4/SEG0 至 COM7/SEG3	18-F		
V_{LC0} 至 V_{LC2}	–	–	
$\overline{RESET}$	2	输入	直接连接或通过电阻连接至 V_{DD}。
FLMD0	38		连接至 V_{SS} [注3] 。
AV_{REF} [注2]	–	–	直接连接至 V_{DD}。[注4]
AV_{SS} [注2]			直接连接 V_{SS} 。

注 **1.** 这些引脚不使用时，在输入/输出端口模式下使用上述推荐连接（参见图 **5-2 时钟操作模式选择寄存器（OSCCTL）的格式**）。

2. 仅限μPD78F045x 和 78F046x 。

3. 当 flash 存储器写入数据时，使用 FLMD0 引脚。当在线重写 flash 存储器数据或执行片上调试，通过电阻将该引脚连接到 V_{SS}（10 kΩ：推荐）。

4. 当端口 2 作为数字端口使用时，应该使它与 V_{DD} 引脚的电平相同。

图 **2-1.** 引脚 输入/输出 电路列表（**1/2**）

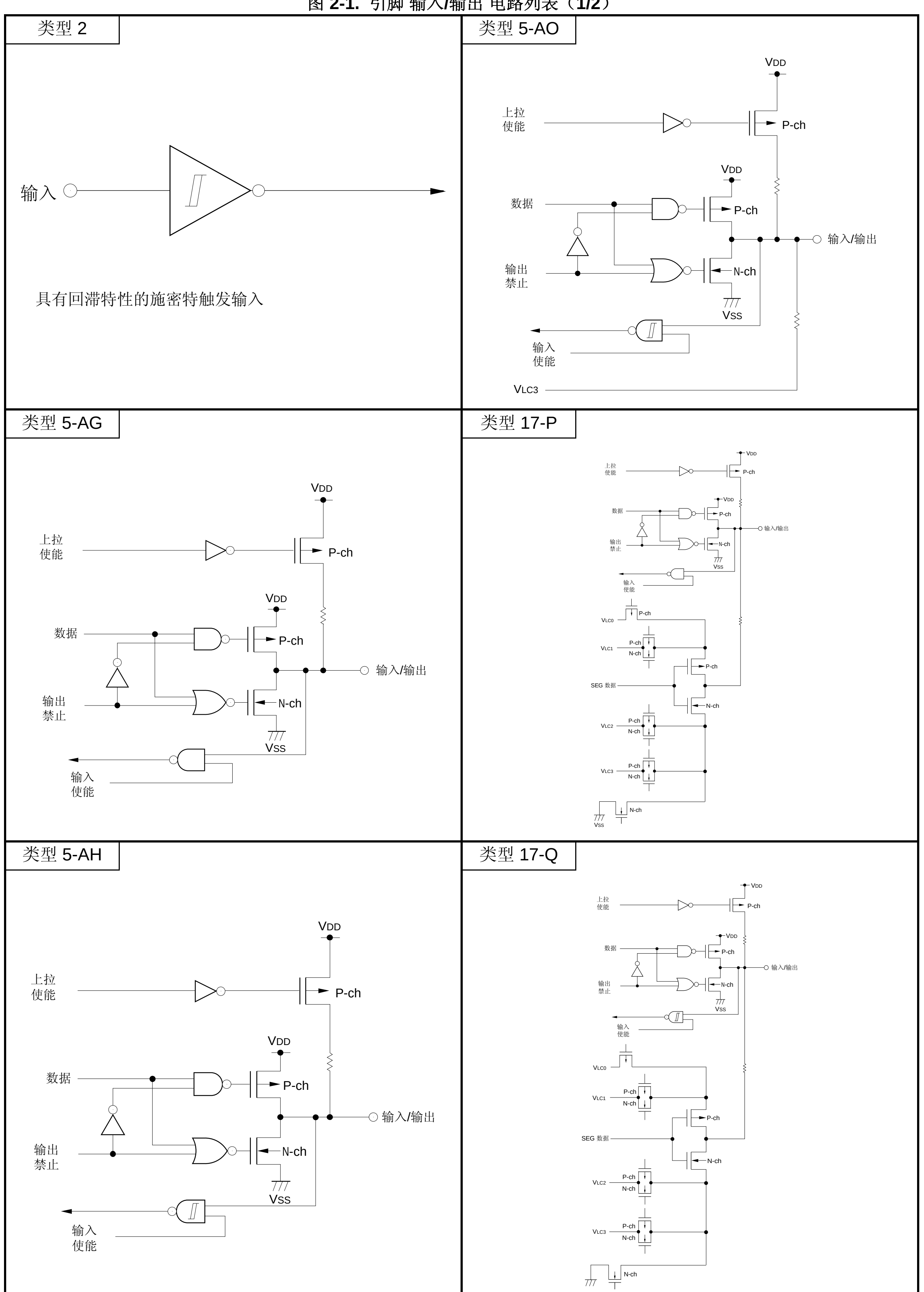
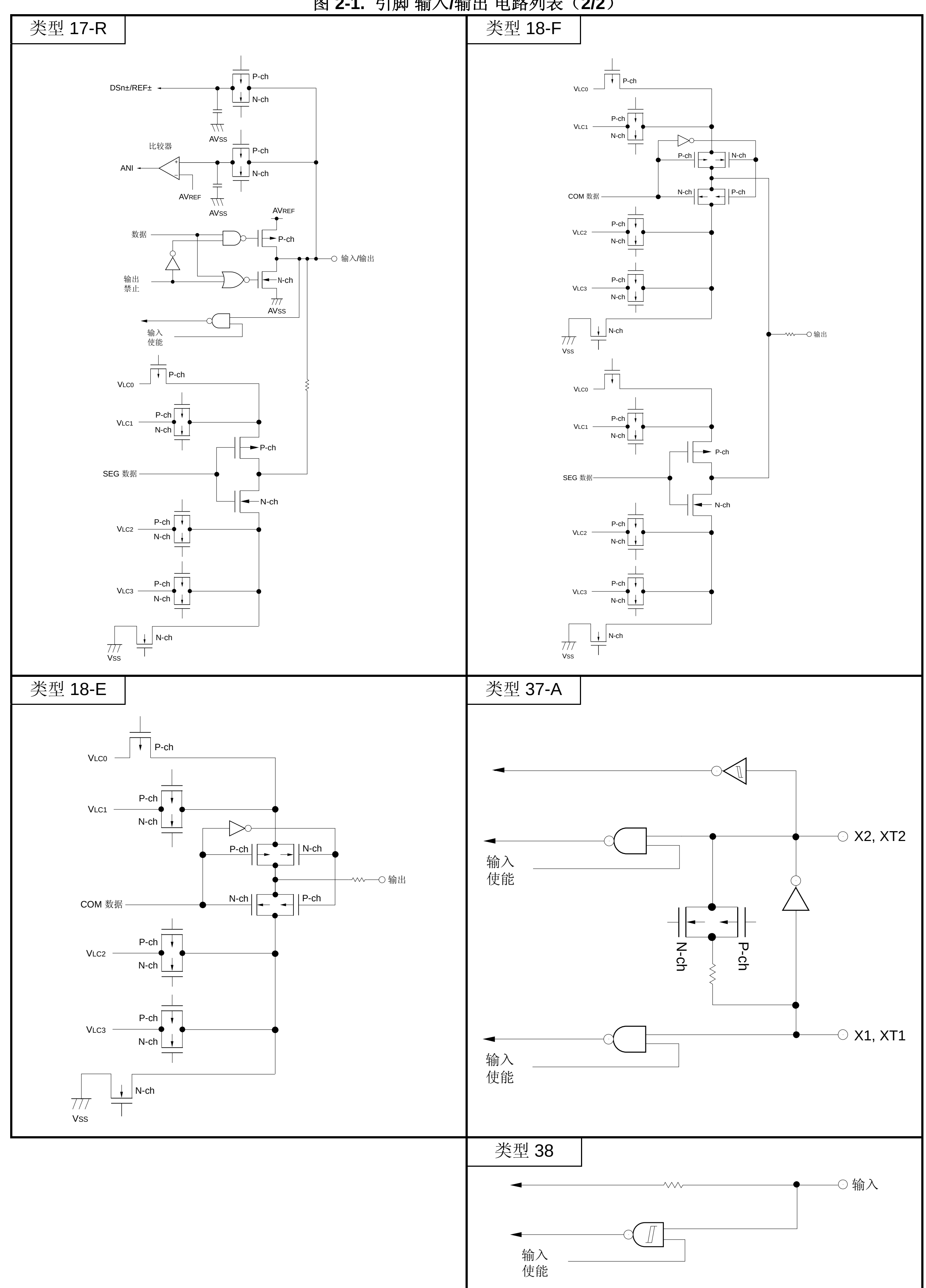


图 2-1. 引脚 输入/输出 电路列表（2/2）



3.1 存储空间

78K0/ LE3 产品可以访问大小为 64KB 的存储空间。图 3-1 至 3-10 显示了存储空间映射图。

注意事项：不论内部存储器容量的大小，所有 78K0/ LE3 产品的内部存储器容量切换寄存器（IMS）和内部扩展 RAM 容量切换寄存器（IXS）的初始值都是固定的（IMS = CFH，IXS = 0CH）。因此，请根据下表对每种产品进行相应设置。

表 **3-1** 内存容量切换寄存器（**IMS**）和内部扩展 **RAM** 容量切换寄存器（**IXS**）的设置值

Flash 存储器版本（78K0/LE3）	IMS	IXS	ROM 容量	内部高速 RAM 容量	内部扩展 RAM 容量
μPD78F0441，78F0451，78F0461	04H	0CH	16 KB	768 字节	–
μPD78F0442，78F0452，78F0462	C6H		24 KB	1 KB	
μPD78F0443，78F0453，78F0463	C8H		32 KB		
μPD78F0444，78F0454，78F0464	CCH	0AH	48 KB		1 KB
μPD78F0445，78F0455，78F0465	CFH		60 KB		

图 3-1. 存储空间映射图（μPD78F0441，78F0451）

FFFFH
特殊功能寄存器
(SFR)
256 x 8 位
FF00H
FEFFH
通用寄存器
32 x 8 位
FEE0H
FEDFH
内部高速 RAM
768 x 8 位
FC00H
FBFFH
保留
FA60H
FA5FH
LCD 显示 RAM
32 × 8 位
FA40H
FA3FH
保留
4000H
3FFFH
Flash 存储器
16384 x 8 位
0000H

数据
存储器空间

程序
存储器空间

3FFFH
程序区域
108FH
108EH
片上调试安全ID
设置区域[注 1]
10 × 8 位
1085H
1084H
选项字节区域[注 1]
5 × 8 位
1080H
107FH
程序区域
1000H
0FFFH
CALLF 入口区域
2048 × 8 位
0800H
07FFH
程序区域
1905 × 8 位
008FH
008EH
片上调试安全ID
设置区域[注 1]
10 × 8 位
0085H
0084H
选项字节区域[注 1]
5 × 8 位
0080H
007FH
CALLT 表区域
64 × 8 位
0040H
003FH
向量表区域
64 × 8 位
0000H

1FFFH
引导簇 1
引导簇0[注2]

注 1. 不使用引导交换时：在 0080H 至 0084H 处设置选项字节，在 0085H 至 008EH 处设置片上调试安全 ID。

当使用引导交换时：在 0080H 至 0084H 和 1080H 至 1084H 处设置选项字节，在 0085H 至 008EH 和 1085H 至 108EH 处设置片上调试安全 ID。

2. 根据安全设置，可以禁止对引导簇 0 的写入（参见 **27.8 安全设置**）。

备注 Flash 存储器被分割成块（1 个 块 = 1 KB）。关于地址值和块编号，参见 **表 3-2 Flash 存储器中地址值和块编号的对应关系。**

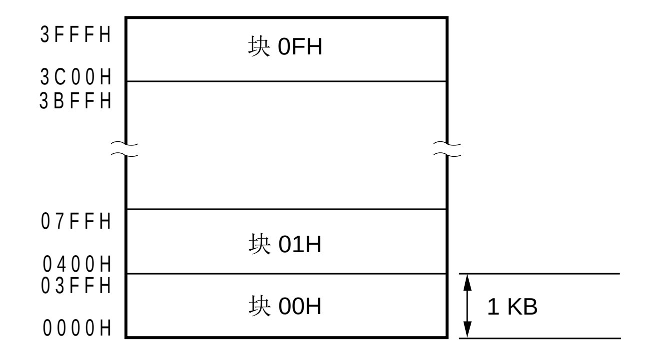


图 3-2. 存储器映射图（μPD78F0461）

FFFFH
特殊功能寄存器
(SFR)
256 x 8 位
FF00H
FEFFH
通用寄存器
32 x 8 位
FEE0H
FEDFH
内部高速 RAM
768 x 8 位
FC00H
FBFFH
保留
FA58H
FA57H
LCD 显示 RAM
24 × 8 位
FA40H
FA3FH
保留[注 3]
4000H
3FFFH
Flash 存储器
16384 x 8 位
0000H

数据
存储器空间

程序
存储器空间

3FFFH
程序区域
108FH
108EH
片上调试安全ID
设置区域[注 1]
10 × 8 位
1085H
1084H
选项字节区域[注 1]
5 × 8 位
1080H
107FH
程序区域
1000H
0FFFH
CALLF 入口区域
2048 × 8 位
0800H
07FFH
程序区域
1905 × 8 位
008FH
008EH
片上调试安全ID
设置区域[注 1]
10 × 8 位
0085H
0084H
选项字节区域[注 1]
5 × 8 位
0080H
007FH
CALLT 表区域
64 × 8 位
0040H
003FH
向量表区域
64 × 8 位
0000H

1FFFH
引导簇 1
引导簇0[注2]

注 **1.** 不使用引导交换时： 在 0080H 至 0084H 处设置选项字节，在 0085H 至 008EH 处设置片上调试安全 ID。

当使用引导交换时： 在 0080H 至 0084H 和 1080H 至 1084H 处设置选项字节，在 0085H 至 008EH 和 1085H 至 108EH 处设置片上调试安全 ID。

2. 根据安全设置，可以禁止对引导簇 0 的写入（参见 **27.8 安全设置**）。

<R> **3.** 但是，可以使用 FA26H 和 FA27H（参见 **13.3 16-位 ΔΣ 型 A/D 转换器中使用的寄存器**）.

备注 Flash 存储器被分割成块（1 个 块 = 1 KB）。关于地址值和块编号，参见 **表 3-2 Flash 存储器中地址值和块编号的对应关系。**

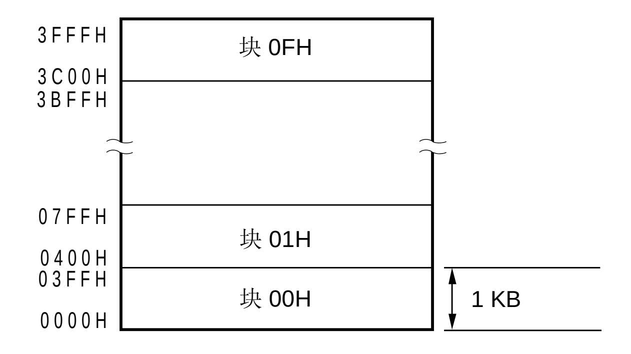


图 3-3. 存储器映射图（*μ*PD78F0442，78F0452）

地址	区域	
FFFFH – FF00H	特殊功能寄存器 (SFR) 256 x 8 位	数据存储器空间
FEFFH – FEE0H	通用寄存器 32 x 8 位	
FEDFH – FB00H	内部高速 RAM 1024 x 8 位	
FAFFH – FA60H	保留	
FA5FH – FA40H	LCD 显示 RAM 32 × 8 位	
FA3FH – 6000H	保留	
5FFFH – 0000H	Flash 存储器 24576 x 8 位	程序存储器空间

地址	区域	
5FFFH – 108FH	程序区域	
108EH – 1085H	片上调试安全ID 设置区域注 1 10 × 8 位	引导簇 1 (1FFFH – 1000H)
1084H – 1080H	选项字节区域注 1 5 × 8 位	
107FH – 1000H	程序区域	
0FFFH – 0800H	CALLF 入口区域 2048 × 8 位	引导簇0注2 (0FFFH – 0000H)
07FFH – 008FH	程序区域 1905 × 8 位	
008EH – 0085H	片上调试安全ID 设置区域注 1 10 × 8 位	
0084H – 0080H	选项字节区域注 1 5 × 8 位	
007FH – 0040H	CALLT 表区域 64 × 8 位	
003FH – 0000H	向量表区域 64 × 8 位	

注 **1.** 不使用引导交换时：在 0080H 至 0084H 处设置选项字节，在 0085H 至 008EH 处设置片上调试安全 ID。

当使用引导交换时：在 0080H 至 0084H 和 1080H 至 1084H 处设置选项字节，在 0085H 至 008EH 和 1085H 至 108EH 处设置片上调试安全 ID。

2. 根据安全设置，可以禁止对引导簇 0 的写入（参见 **27.8 安全设置**）。

备注 Flash 存储器被分割成块（1 个 块 = 1 KB）。关于地址值和块编号，参见 表 **3-2 Flash 存储器中地址值和块编号的对应关系。**

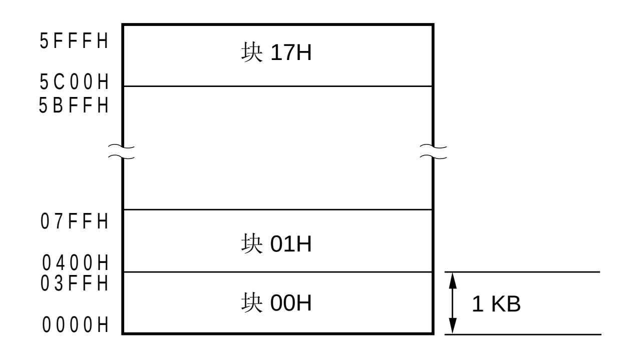


图 3-4. 存储空间映射图（μPD78F0462）

FFFFH
特殊功能寄存器
(SFR)
256 x 8 位
FF00H
FEFFH
通用寄存器
32 x 8 位
FEE0H
FEDFH
内部高速 RAM
1024 x 8 位
FB00H
FAFFH
保留
FA58H
FA57H
LCD 显示 RAM
24 × 8 位
FA40H
FA3FH
保留[注 3]
6000H
5FFFH
Flash 存储器
24576 x 8 位
0000H

数据
存储器空间

程序
存储器空间

5FFFH
程序区域
108FH
108EH
片上调试安全ID
设置区域[注 1]
10 × 8 位
1085H
1084H
选项字节区域[注 1]
5 × 8 位
1080H
107FH
程序区域
1000H
0FFFH
CALLF 入口区域
2048 × 8 位
0800H
07FFH
程序区域
1905 × 8 位
008FH
008EH
片上调试安全ID
设置区域[注 1]
10 × 8 位
0085H
0084H
选项字节区域[注 1]
5 × 8 位
0080H
007FH
CALLT 表区域
64 × 8 位
0040H
003FH
向量表区域
64 × 8 位
0000H

1FFFH
引导簇 1
引导簇0[注2]

注 **1.** 不使用引导交换时： 在 0080H 至 0084H 处设置选项字节，在 0085H 至 008EH 处设置片上调试安全 ID。

当使用引导交换时： 在 0080H 至 0084H 和 1080H 至 1084H 处设置选项字节，在 0085H 至 008EH 和 1085H 至 108EH 处设置片上调试安全 ID。

2. 根据安全设置，可以禁止对引导簇 0 的写入（参见 **27.8 安全设置**）。

<R> **3.** 但是，可以使用 FA26H 和 FA27H（参见 **13.3 16-位 ΔΣ 型 A/D 转换器中使用的寄存器**）.

备注 Flash 存储器被分割成块（1 个 块 = 1 KB）。关于地址值和块编号，参见 **表 3-2 Flash 存储器中地址值和块编号的对应关系。**

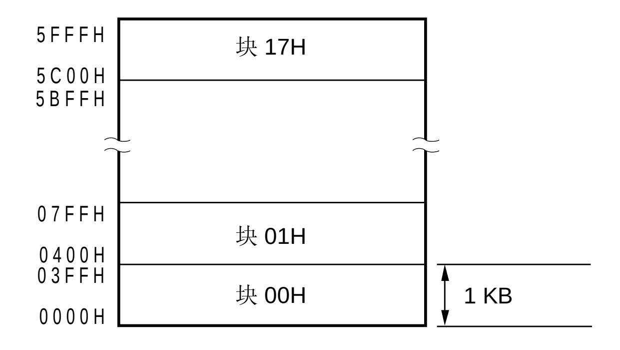


图 3-5. 存储空间映射图（μPD78F0443，78F0453）

FFFFH
特殊功能寄存器
(SFR)
256 x 8 位
FF00H
FEFFH
通用寄存器
32 x 8 位
FEE0H
FEDFH
内部高速 RAM
1024 x 8 位
FB00H
FAFFH
保留
FA60H
FA5FH
LCD 显示 RAM
32 × 8 位
FA40H
FA3FH
保留
8000H
7FFFH
Flash 存储器
32768 x 8 位
0000H

数据
存储器空间

程序
存储器空间

7FFFH
程序区域
1FFFH
108FH
108EH
片上调试安全ID
设置区域注 1
10 × 8 位
1085H
1084H
选项字节区域注 1
5 × 8 位
1080H
107FH
程序区域
1000H
0FFFH
引导簇 1
CALLF 入口区域
2048 × 8 位
0800H
07FFH
程序区域
1905 × 8 位
008FH
008EH
片上调试安全ID
设置区域注 1
10 × 8 位
0085H
0084H
选项字节区域注 1
5 × 8 位
0080H
007FH
CALLT 表区域
64 × 8 位
0040H
003FH
向量表区域
64 × 8 位
0000H
引导簇0注2

注 1. 不使用引导交换时：在 0080H 至 0084H 处设置选项字节，在 0085H 至 008EH 处设置片上调试安全 ID。

当使用引导交换时：在 0080H 至 0084H 和 1080H 至 1084H 处设置选项字节，在 0085H 至 008EH 和 1085H 至 108EH 处设置片上调试安全 ID。

2. 根据安全设置，可以禁止对引导簇 0 的写入（参见 **27.8 安全设置**）。

备注 Flash 存储器被分割成块（1 个 块 = 1 KB）。关于地址值和块编号，参见 **表 3-2 Flash 存储器中地址值和块编号的对应关系。**

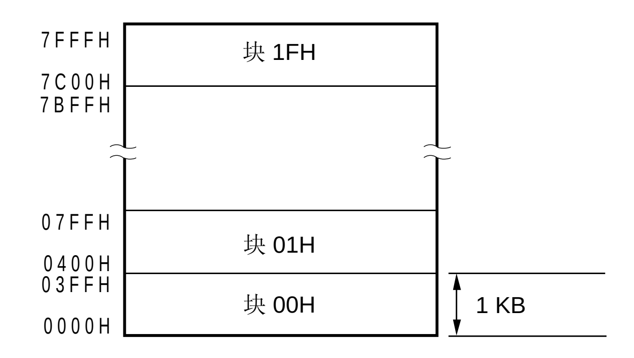


图 **3-6.** 存储空间映射图（*μ*PD78F0463）

地址	区域
FFFFH – FF00H	特殊功能寄存器 (SFR) 256 x 8 位
FEFFH – FEE0H	通用寄存器 32 x 8 位
FEDFH – FB00H	内部高速 RAM 1024 x 8 位
FAFFH – FA58H	保留
FA57H – FA40H	LCD 显示 RAM 24 × 8 位
FA3FH – 8000H	保留[注 3]
7FFFH – 0000H	Flash 存储器 32768 x 8 位

数据存储器空间：FFFFH – 0000H

程序存储器空间：7FFFH – 0000H

地址	区域	
7FFFH – 108FH	程序区域	引导簇 1（1FFFH – 1000H）
108EH – 1085H	片上调试安全ID 设置区域[注 1] 10 × 8 位	
1084H – 1080H	选项字节区域[注 1] 5 × 8 位	
107FH – 1000H	程序区域	
0FFFH – 0800H	CALLF 入口区域 2048 × 8 位	引导簇0[注2]（0FFFH – 0000H）
07FFH – 008FH	程序区域 1905 × 8 位	
008EH – 0085H	片上调试安全ID 设置区域[注 1] 10 × 8 位	
0084H – 0080H	选项字节区域[注 1] 5 × 8 位	
007FH – 0040H	CALLT 表区域 64 × 8 位	
003FH – 0000H	向量表区域 64 × 8 位	

注 **1.** 不使用引导交换时： 在 0080H 至 0084H 处设置选项字节，在 0085H 至 008EH 处设置片上调试安全 ID。

当使用引导交换时： 在 0080H 至 0084H 和 1080H 至 1084H 处设置选项字节，在 0085H 至 008EH 和 1085H 至 108EH 处设置片上调试安全 ID。

2. 根据安全设置，可以禁止对引导簇 0 的写入（参见 **27.8 安全设置**）。

<R> **3.** 但是，可以使用 FA26H 和 FA27H（参见 **13.3 16-位 ΔΣ 型 A/D 转换器中使用的寄存器**）.

备注 Flash 存储器被分割成块（1 个 块 = 1 KB）。关于地址值和块编号，参见 **表 3-2 Flash 存储器中地址值和块编号的对应关系。**

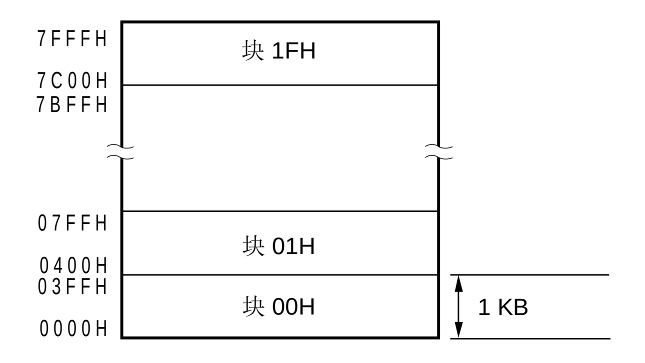


图 **3-7. 存储空间映射图（μPD78F0444，78F0454）**

地址	区域
FFFFH – FF00H	特殊功能寄存器 (SFR) 256 × 8 位
FEFFH – FEE0H	通用寄存器 32 × 8 位
FEDFH – FB00H	内部高速 RAM 1024 × 8 位
FAFFH – FA60H	保留
FA5FH – FA40H	LCD 显示 RAM 32 × 8 位
FA3FH – F800H	保留
F7FFH – F400H	内部扩展 RAM 1024 × 8 位
F3FFH – C000H	保留
BFFFH – 0000H	Flash 存储器 49152 × 8 位

数据存储器空间：FFFFH – 0000H

程序 RAM 区域（可以取指的 RAM空间）：F7FFH – F400H

程序存储器空间：BFFFH – 0000H

地址	区域	
BFFFH – 108FH	程序区域	
108EH – 1085H	片上调试安全ID 设置区域[注 1] 10 × 8 位	引导簇 1（1FFFH – 1000H）
1084H – 1080H	选项字节区域[注 1] 5 × 8 位	
107FH – 1000H	程序区域	
0FFFH – 0800H	CALLF 入口区域 2048 × 8 位	引导簇0[注2]（0FFFH – 0000H）
07FFH – 008FH	程序区域 1905 × 8 位	
008EH – 0085H	片上调试安全ID 设置区域[注 1] 10 × 8 位	
0084H – 0080H	选项字节区域[注 1] 5 × 8 位	
007FH – 0040H	CALLT 表区域 64 × 8 位	
003FH – 0000H	向量表区域 64 × 8 位	

注 1. 不使用引导交换时：在 0080H 至 0084H 处设置选项字节，在 0085H 至 008EH 处设置片上调试安全 ID。

当使用引导交换时：在 0080H 至 0084H 和 1080H 至 1084H 处设置选项字节，在 0085H 至 008EH 和 1085H 至 108EH 处设置片上调试安全 ID。

2. 根据安全设置，可以禁止对引导簇 0 的写入（参见 **27.8 安全设置**）。

备注 Flash 存储器被分割成块（1 个 块 = 1 KB）。关于地址值和块编号，参见 **表 3-2 Flash 存储器中地址值和块编号的对应关系**。

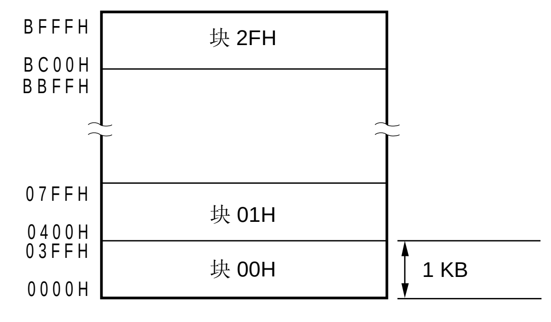


图 3-8. 存储空间映射图（μPD78F0464）

FFFFH
特殊功能寄存器
(SFR)
256 × 8 位
FF00H
FEFFH
通用寄存器
32 × 8 位
FEE0H
FEDFH
内部高速 RAM
1024 × 8 位
FB00H
FAFFH
保留
FA58H
FA57H
LCD 显示 RAM
24 × 8 位
FA40H
FA3FH
保留注 3
F800H
F7FFH
内部扩展 RAM
1024 × 8 位
F400H
F3FFH
保留
C000H
BFFFH
Flash 存储器
49152 × 8 位
0000H

数据
存储器空间

程序 RAM 区域
（可以取指的
RAM空间）

程序
存储器空间

BFFFH
程序区域
108FH
108EH
片上调试安全ID
设置区域注 1
10 × 8 位
1085H
1084H
选项字节区域注 1
5 × 8 位
1080H
107FH
程序区域
1000H
0FFFH
CALLF 入口区域
2048 × 8 位
0800H
07FFH
程序区域
1905 × 8 位
008FH
008EH
片上调试安全ID
设置区域注 1
10 × 8 位
0085H
0084H
选项字节区域注 1
5 × 8 位
0080H
007FH
CALLT 表区域
64 × 8 位
0040H
003FH
向量表区域
64 × 8 位
0000H

1FFFH
引导簇 1
引导簇0注2

注 **1.** 不使用引导交换时：在 0080H 至 0084H 处设置选项字节，在 0085H 至 008EH 处设置片上调试安全 ID。

当使用引导交换时：在 0080H 至 0084H 和 1080H 至 1084H 处设置选项字节，在 0085H 至 008EH 和 1085H 至 108EH 处设置片上调试安全 ID。

2. 根据安全设置，可以禁止对引导簇 0 的写入（参见 **27.8 安全设置**）。

<R> **3.** 但是，可以使用 FA26H 和 FA27H（参见 **13.3 16-位 ΔΣ 型 A/D 转换器中使用的寄存器**）.

备注 Flash 存储器被分割成块（1 个 块 = 1 KB）。关于地址值和块编号，参见 **表 3-2 Flash 存储器中地址值和块编号的对应关系。**

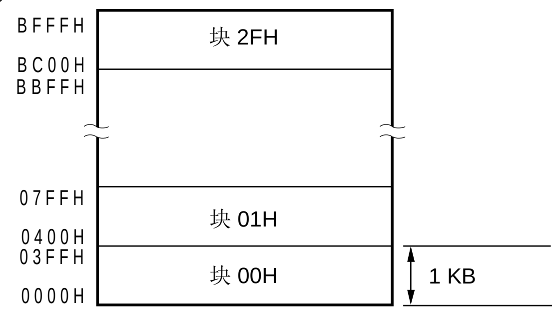


图 3-9. 存储空间映射图（μPD78F0445，78F0455）

地址	区域
FFFFH – FF00H	特殊功能寄存器 (SFR) 256 × 8 位
FEFFH – FEE0H	通用寄存器 32 × 8 位
FEDFH – FB00H	内部高速 RAM 1024 × 8 位
FAFFH – FA60H	保留
FA5FH – FA40H	LCD 显示 RAM 32 × 8 位
FA3FH – F800H	保留
F7FFH – F400H	内部扩展 RAM 1024 × 8 位
F3FFH – F000H	保留
EFFFH – 0000H	Flash 存储器 61440 × 8 位

数据存储器空间：FFFFH – 0000H

程序 RAM 区域（可以取指的 RAM空间）：F7FFH – F400H

程序存储器空间：EFFFH – 0000H

地址	区域	
EFFFH – 108FH	程序区域	
108EH – 1085H	片上调试安全ID 设置区域[注 1] 10 × 8 位	引导簇 1（1FFFH – 1000H）
1084H – 1080H	选项字节区域[注 1] 5 × 8 位	引导簇 1
107FH – 1000H	程序区域	引导簇 1
0FFFH – 0800H	CALLF 入口区域 2048 × 8 位	引导簇0[注2]
07FFH – 008FH	程序区域 1905 × 8 位	引导簇0[注2]
008EH – 0085H	片上调试安全ID 设置区域[注 1] 10 × 8 位	引导簇0[注2]
0084H – 0080H	选项字节区域[注 1] 5 × 8 位	引导簇0[注2]
007FH – 0040H	CALLT 表区域 64 × 8 位	引导簇0[注2]
003FH – 0000H	向量表区域 64 × 8 位	引导簇0[注2]

注 **1.** 不使用引导交换时：在 0080H 至 0084H 处设置选项字节，在 0085H 至 008EH 处设置片上调试安全 ID。

当使用引导交换时：在 0080H 至 0084H 和 1080H 至 1084H 处设置选项字节，在 0085H 至 008EH 和 1085H 至 108EH 处设置片上调试安全 ID。

2. 根据安全设置，可以禁止对引导簇 0 的写入（参见 **27.8 安全设置**）。

备注 Flash 存储器被分割成块（1 个 块 = 1 KB）。关于地址值和块编号，参见 **表 3-2 Flash 存储器中地址值和块编号的对应关系**。

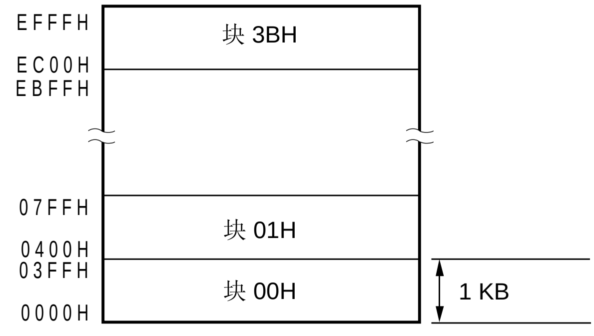


图 **3-10.** 存储空间映射图（**μPD78F0465**）

FFFFH
特殊功能寄存器
(SFR)
256 × 8 位
FF00H
FEFFH
通用寄存器
32 × 8 位
FEE0H
FEDFH
内部高速 RAM
1024 × 8 位
FB00H
FAFFH
保留
FA58H
FA57H
LCD 显示 RAM
24 × 8 位
FA40H
FA3FH
保留注 3
数据
存储器空间
F800H
F7FFH
程序 RAM 区域
（可以取指的
RAM空间）
内部扩展 RAM
1024 × 8 位
F400H
F3FFH
保留
F000H
EFFFH
程序
存储器空间
Flash 存储器
61440 × 8 位
0000H

EFFFH
程序区域
108FH
108EH
片上调试安全ID
设置区域注 1
10 × 8 位
1085H
1084H
选项字节区域注 1
5 × 8 位
1080H
107FH
程序区域
1000H
0FFFH
CALLF 入口区域
2048 × 8 位
0800H
07FFH
程序区域
1905 × 8 位
008FH
008EH
片上调试安全ID
设置区域注 1
10 × 8 位
0085H
0084H
选项字节区域注 1
5 × 8 位
0080H
007FH
CALLT 表区域
64 × 8 位
0040H
003FH
向量表区域
64 × 8 位
0000H

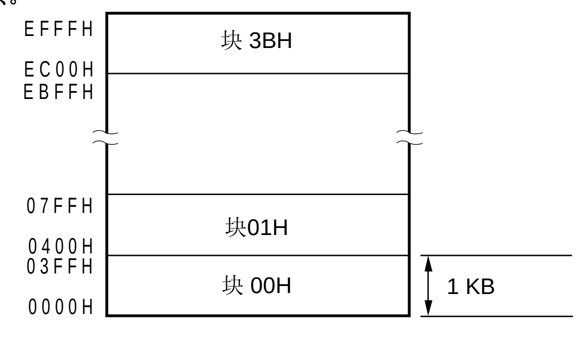
1FFFH
引导簇 1
引导簇0注2

注		
1.	不使用引导交换时：	在 0080H 至 0084H 处设置选项字节，在 0085H 至 008EH 处设置片上调试安全 ID。
	当使用引导交换时：	在 0080H 至 0084H 和 1080H 至 1084H 处设置选项字节，在 0085H 至 008EH 和 1085H 至 108EH 处设置片上调试安全 ID。

2. 根据安全设置，可以禁止对引导簇 0 的写入（参见 **27.8 安全设置**）。

<R> **3.** 但是，可以使用 FA26H 和 FA27H（参见 **13.3 16-**位 ΔΣ 型 **A/D** 转换器中使用的寄存器）.

备注 Flash 存储器被分割成块（1 个 块 = 1 KB）。关于地址值和块编号，参见 表 **3-2 Flash** 存储器中地址值和块编号的对应关系。



Flash 存储器中地址值和块编号之间的关系如下所示。

表 3-2. Flash 存储器中地址值和块编号的对应

地址值	块编号	地址值	块编号	地址值	块编号	地址值	块编号
0000H至03FFH	00H	4000H至43FFH	10H	8000H至83FFH	20H	C000H至C3FFH	30H
0400H至07FFH	01H	4400H至47FFH	11H	8400H至87FFH	21H	C400H至C7FFH	31H
0800H至0BFFH	02H	4800H至4BFFH	12H	8800H至8BFFH	22H	C800H至CBFFH	32H
0C00H至0FFFH	03H	4C00H至4FFFH	13H	8C00H至8FFFH	23H	CC00H至CFFFH	33H
1000H至13FFH	04H	5000H至53FFH	14H	9000H至93FFH	24H	D000H至D3FFH	34H
1400H至17FFH	05H	5400H至57FFH	15H	9400H至97FFH	25H	D400H至D7FFH	35H
1800H至1BFFH	06H	5800H至5BFFH	16H	9800H至9BFFH	26H	D800H至DBFFH	36H
1C00H至1FFFH	07H	5C00H至5FFFH	17H	9C00H至9FFFH	27H	DC00H至DFFFH	37H
2000H至23FFH	08H	6000H至63FFH	18H	A000H至A3FFH	28H	E000H至E3FFH	38H
2400H至27FFH	09H	6400H至67FFH	19H	A400H至A7FFH	29H	E400H至E7FFH	39H
2800H至2BFFH	0AH	6800H至6BFFH	1AH	A800H至ABFFH	2AH	E800H至EBFFH	3AH
2C00H至2FFFH	0BH	6C00H至6FFFH	1BH	AC00H至AFFFH	2BH	EC00H至EFFFH	3BH
3000H至33FFH	0CH	7000H至73FFH	1CH	B000H至B3FFH	2CH		
3400H至37FFH	0DH	7400H至77FFH	1DH	B400H至B7FFH	2DH		
3800H至3BFFH	0EH	7800H至7BFFH	1EH	B800H至BBFFH	2EH		
3C00H至3FFFH	0FH	7C00H至7FFFH	1FH	BC00H至BFFFH	2FH		

备注 μPD78F0441，78F0451，78F0461：块编号 00H 至 0FH
μPD78F0442，78F0452，78F0462：块编号 00H 至 17H
μPD78F0443，78F0453，78F0463：块编号 00H 至 1FH
μPD78F0444，78F0454，78F0464：块编号 00H 至 2FH
μPD78F0445，78F0455，78F0465：块编号 00H 至 3BH

3.1.1 内部程序存储空间

内部程序存储空间用于存储程序和表数据，一般通过程序计数器（PC）来寻址。

78K0/LE3 产品包括内部 ROM（Flash 存储器），如下表所示。

表 3-3. 内部 ROM 容量

产品型号	内部 ROM	
	结构	容量
μPD78F0441，78F0451，78F0461	Flash 存储器	16384 × 8 位 （0000H 至 3FFFH）
μPD78F0442，78F0452，78F0462		24576 × 8 位 （0000H 至 5FFFH）
μPD78F0443，78F0453，78F0463		32768 × 8 位 （0000H 至 7FFFH）
μPD78F0444，78F0454，78F0464		49152 × 8 位 （0000H 至 BFFFH）
μPD78F0445，78F0455，78F0465		61440 × 8 位 （0000H 至 EFFFH）

内部程序存储空间主要分为以下几个区域。

（1） 向量表区域

从 0000H 到 003FH 总共 64 字节作为向量表区域。在向量表中存放的是根据复位信号输入或每个中断请求的产生进行转移的程序的起始地址。

在 16 位地址中，低 8 位是偶地址，高 8 位是奇地址。

表 3-4. 向量表

向量表地址	中断源	向量表地址	中断源
0000H	$\overline{\text{RESET}}$ 输入，POC, LVI, WDT	0022H	INTTM010
0004H	INTLVI	0024H [注1]	INTAD [注1]
0006H	INTP0	0026H	INTSR0
0008H	INTP1	0028H	INTRTC
000AH	INTP2	002AH	INTTM51
000CH	INTP3	002CH	INTKR
000EH	INTP4	002EH	INTRTCI
0012H	INTSRE6	0030H [注2]	INTDSAD [注2]
0014H	INTSR6	0032H	INTTM52
0016H	INTST6	0034H	INTTMH2
0018H	INTCSI10/INTST0	0036H	INTMCG
001AH	INTTMH1	0038H	INTRIN
001CH	INTTMH0	003AH	INTRERR/INTGP/INTREND /INTDFULL
001EH	INTTM50	003EH	BRK
0020H	INTTM000		

注 **1.** 仅限μPD78F045x 和 78F046x 。

2. 仅限μPD78F046x 。

（2） CALLT 指令表区域

0040H 至 007FH 共 64 字节的区域，可存放 1 字节调用指令（CALLT）的子程序入口地址。

（3） 选项字节区域

0080H 至 0084H 和 1080H 至 1084H 各有 5 字节的区域可以用作选项字节区域。不使用引导交换功能时，在 0080H 至 0084H 处设置选项字节；而当使用引导交换功能时，在 0080H 至 0084H 和 1080H 至 1084H 处设置选项字节。详情参见 **第二十六章 选项字节**。

（4） CALLF 指令入口区域

0800H 至 0FFFH 的区域可以执行子程序的直接调用，通过 2 字节的调用指令（CALLF）实现。

（5） 片上调试安全 ID 设置区域

0085H 至 008EH 和 1085H 至 108EH 各 10 字节区域可用作片上调试安全 ID 设置区域。不使用引导交换功能时，在 0085H 至 008EH 区域设置片上调试安全 ID；而当使用引导交换功能时，在 0085H 至 008EH 和 1085H 至 108EH 区域设置。详情参见**第二十八章 片上调试功能**。

3.1.2 内部数据存储器空间

78K0/LE3 产品包括以下几种 RAM。

（1） 内部高速 RAM

表 3-5. 内部高速 RAM 容量

产品型号	内部高速 RAM
μPD78F0441，78F0451，78F0461	768 × 8 位 （FC00H 至 FEFFH）
μPD78F0442，78F0452，78F0462	1024 × 8 位 （FB00H 至 FEFFH）
μPD78F0443，78F0453，78F0463	
μPD78F0444，78F0454，78F0464	
μPD78F0445，78F0455，78F0465	

该区域不可以用作写入和执行指令的程序区域。

内部高速 RAM 也可以作为堆栈存储器使用。

（2） 内部扩展 RAM

表 3-6. 内部扩展 RAM 容量

产品型号	内部高速 RAM
μPD78F0441，78F0451，78F0461	–
μPD78F0442，78F0452，78F0462	
μPD78F0443，78F0453，78F0463	
μPD78F0444，78F0454，78F0464	1024 × 8 位（F400H 至 F7FFH）
μPD78F0445，78F0455，78F0465	

与内部高速 RAM 类似，内部扩展 RAM 可作为普通数据区域，同时也可作为写入和执行指令的程序存储区域。内部扩展 RAM 不能用作堆栈。

（3） LCD 显示 RAM

LCD 控制器/驱动器内包括 LCD 显示 RAM（参见图 **18-5 LCD 显示 RAM**）。

表 3-7. LCD 显示 RAM 容量

产品型号	内部扩展 RAM
μPD78F044x，78F045x	32× 8 位 （FA40H 至 FA5FH）
μPD78F046x	24× 8 位 （FA40H 至 FA57H）

3.1.3 特殊功能寄存器（SFR）区域

片上外设硬件的特殊功能寄存器（SFR）分配在 FF00H 至 FFFFH 的区域 （参见 **3.2.3 特殊功能寄存器（SFR）**中**表 3-8 特殊功能寄存器列表**）。

注意事项：不要访问那些没有分配特殊功能寄存器的地址。

3.1.4 数据存储空间寻址

寻址是定位地址的方式，需要定位的地址包括将要执行的下一条指令地址或者与指令执行相关的寄存器地址或存储器地址。

基于可操作性和其他考虑，在 78K0/LE3 中提供了几种寻址方式，用于与指令执行相关的存储器寻址。因为有专门的数据存储区域，所以对于特殊功能寄存器（SFR）和通用寄存器来说，可以使用一些特殊的寻址方式。图 3-11 至 3-20 显示了数据存储空间与寻址的对应关系。每种寻址方式的详细信息，参见 **3.4 操作数地址寻址**。

图 3-11. 数据存储空间与寻址方式的对应关系（μPD78F0441，78F0451）

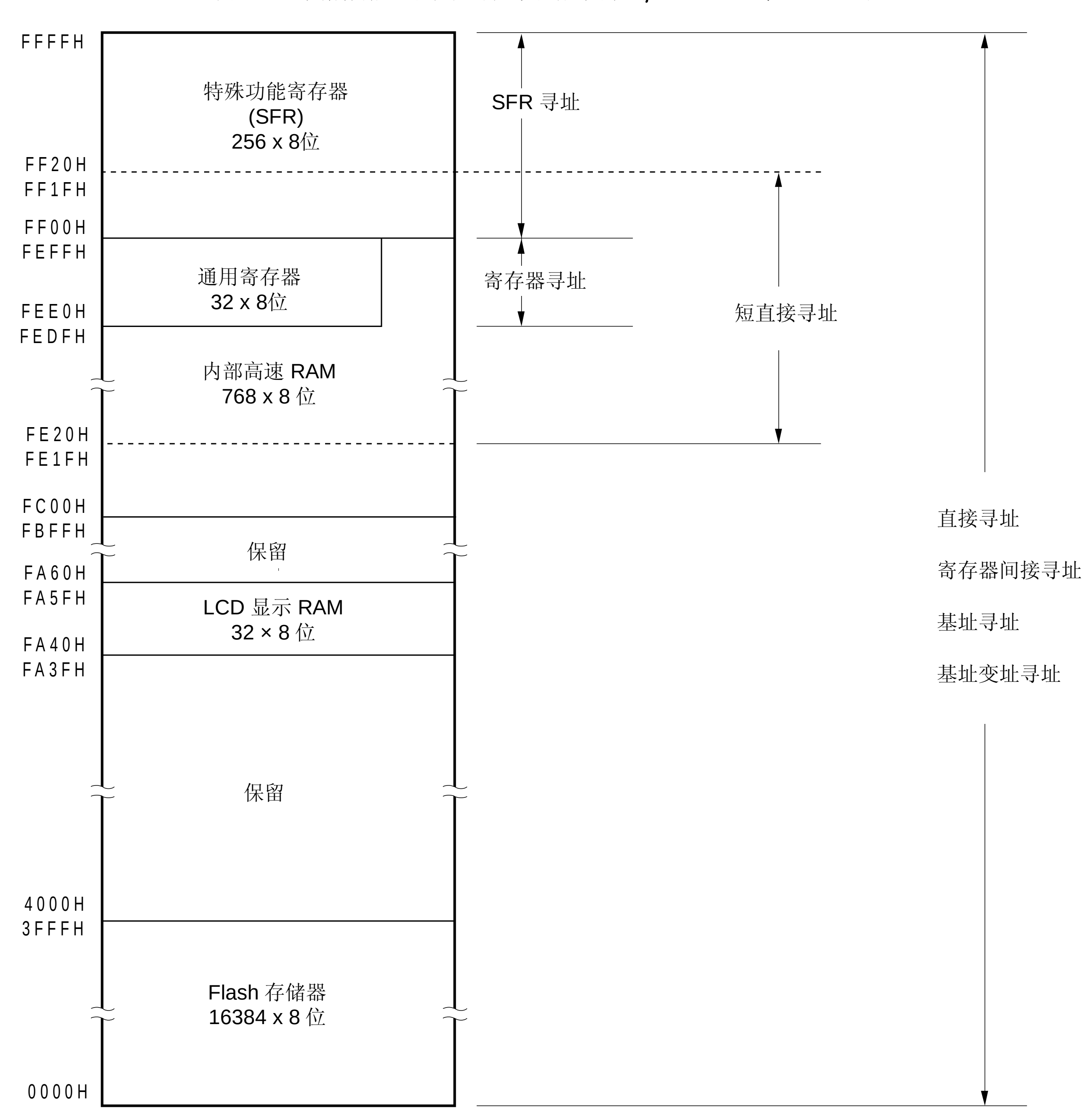
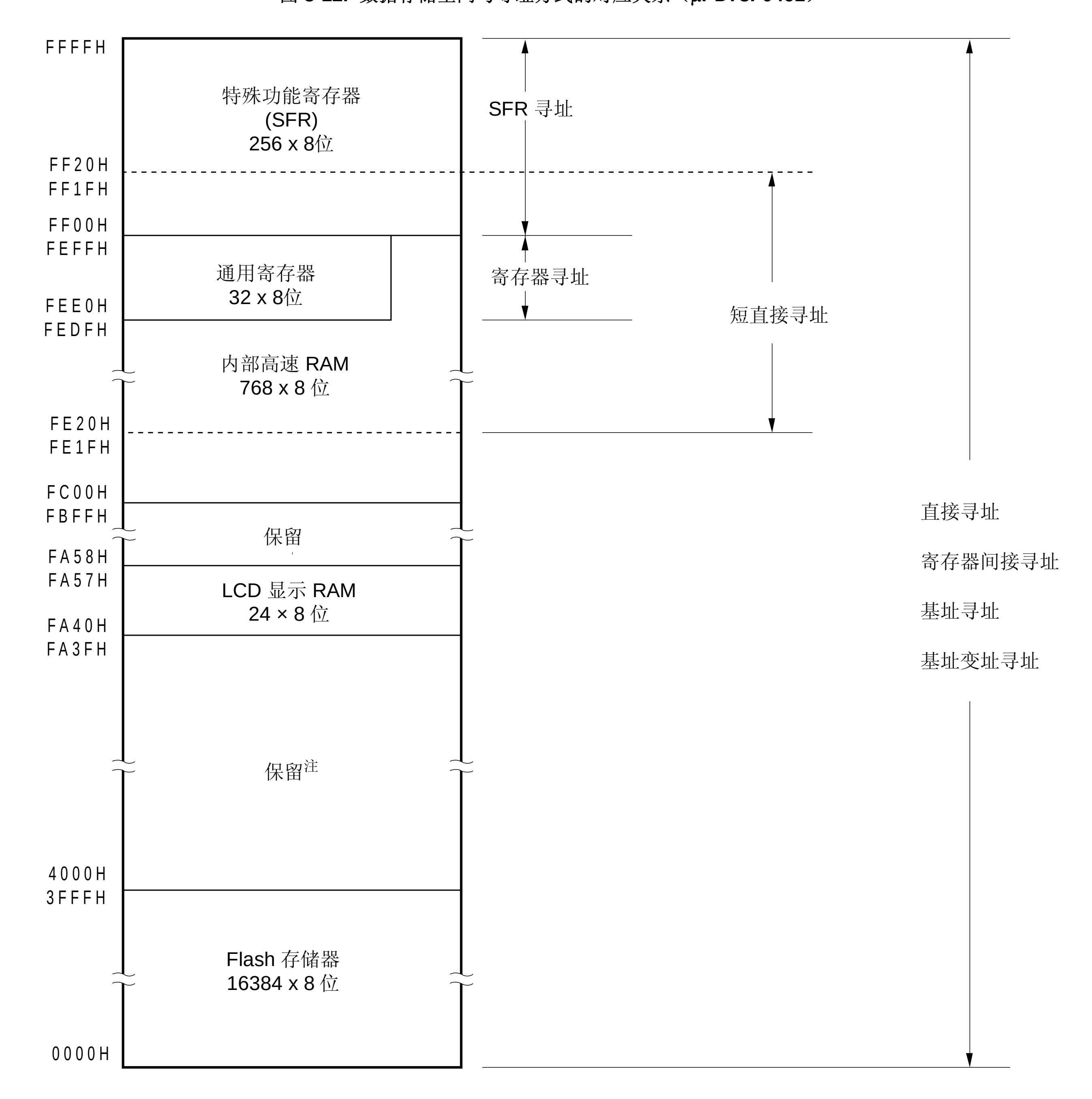


图 **3-12.** 数据存储空间与寻址方式的对应关系（**μPD78F0461**）



<R> 注　但是，可以使用 FA26H 和 FA27H（参见 **13.3 16** 位ΔΣ类型 **A/D** 转换器中使用的寄存器）

图 3-13. 数据存储空间与寻址方式的对应关系（μPD78F0442，F0452）

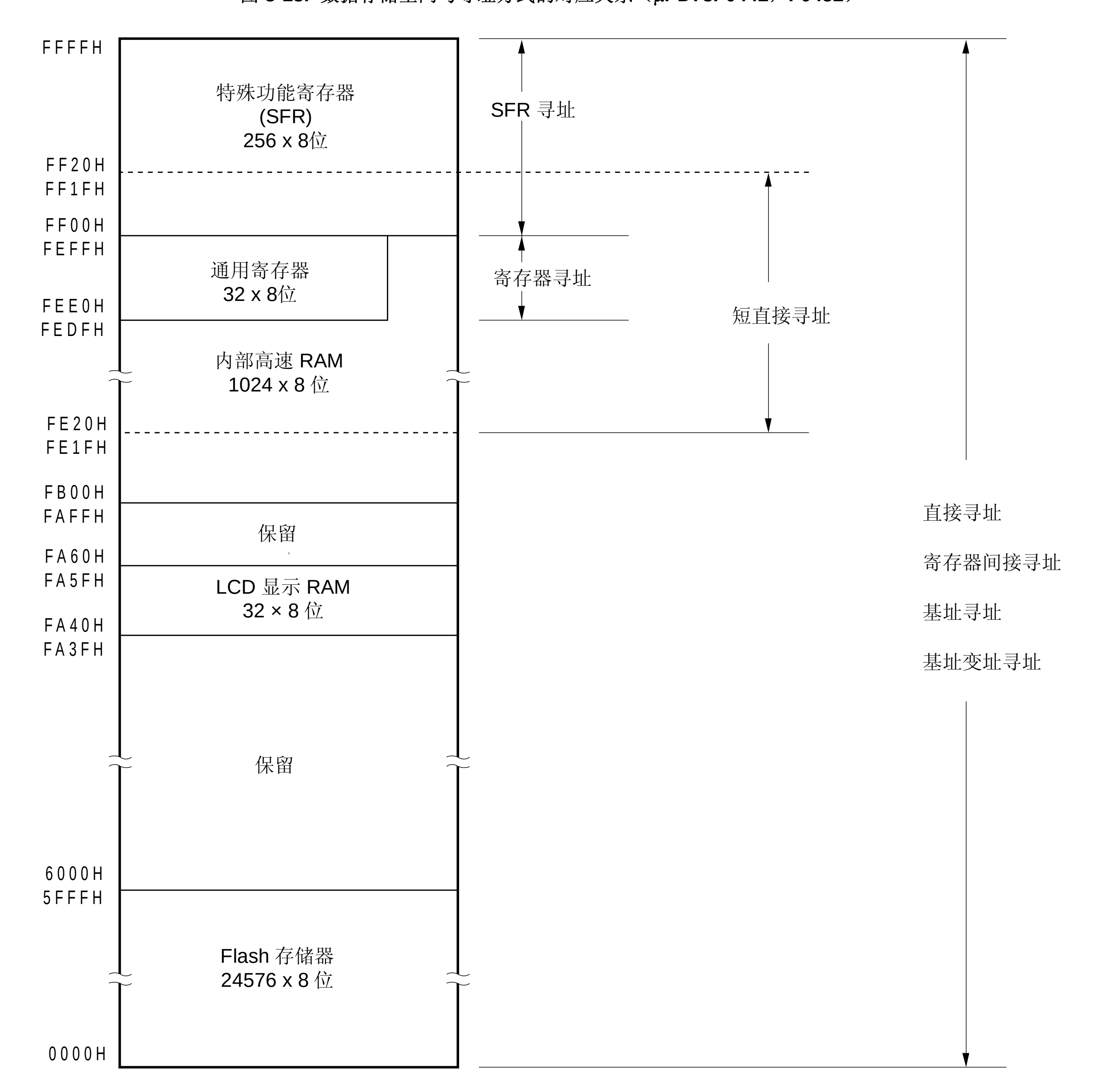
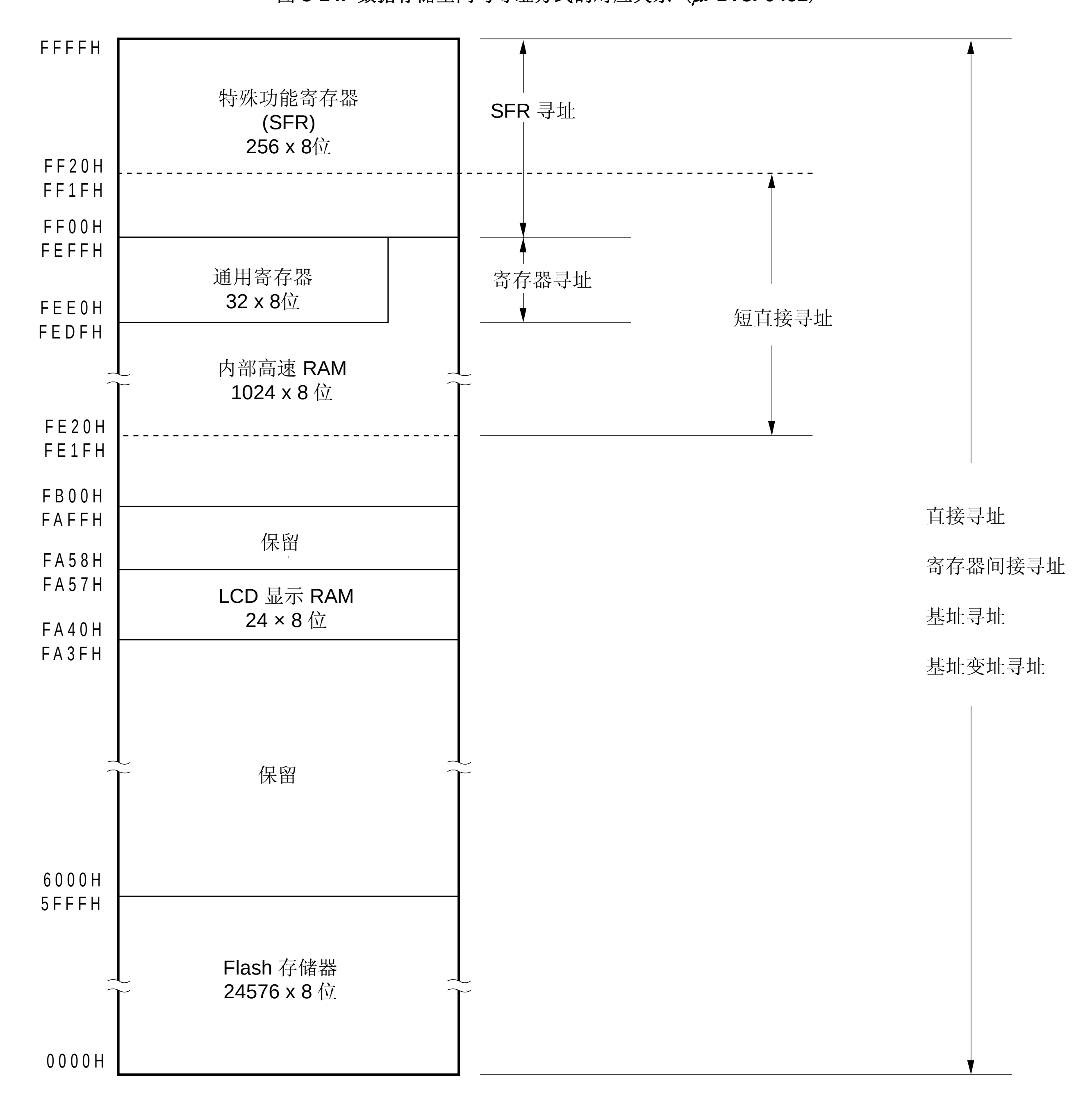


图 3-14. 数据存储空间与寻址方式的对应关系（μPD78F0462）



<R> 注 但是，可以使用 FA26H 和 FA27H（参见 **13.3 16 位ΔΣ类型 A/D 转换器中使用的寄存器**）

图 3-15. 数据存储空间与寻址方式的对应关系（ （μPD78F0443，F0453）

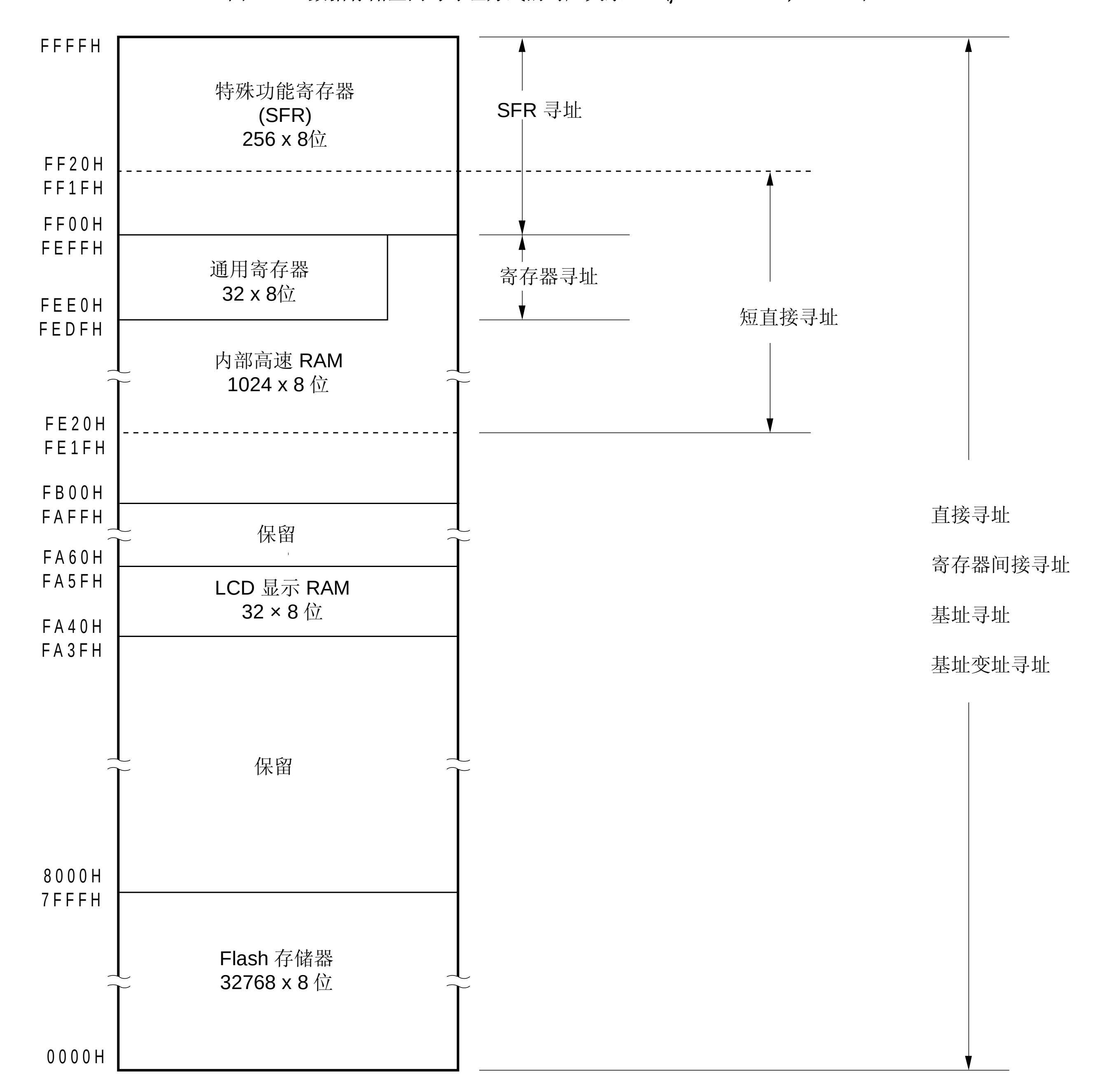
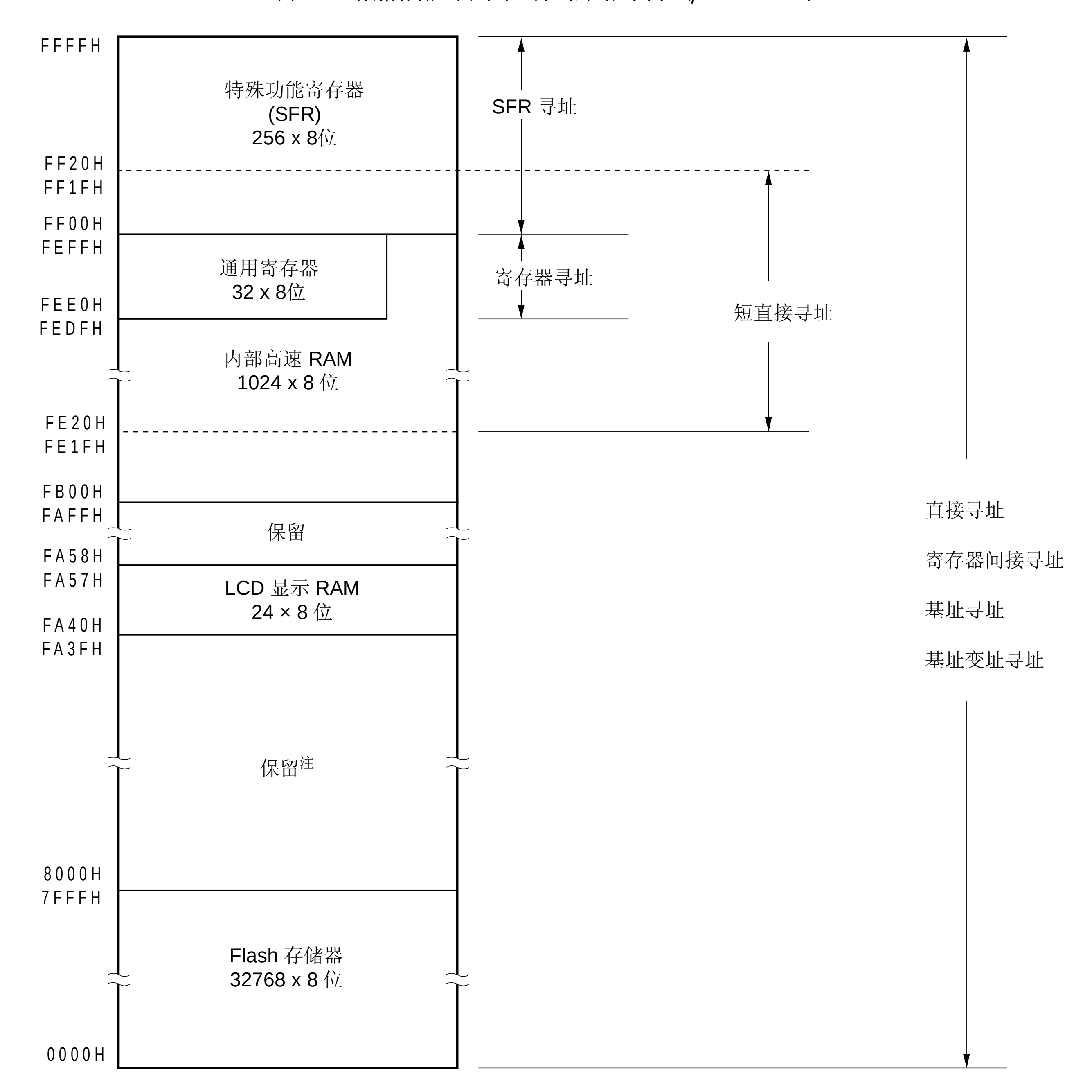


图 **3-16.** 数据存储空间与寻址方式的对应关系（**μPD78F0463**）



<R> 注　但是，可以使用 FA26H 和 FA27H（参见 **13.3 16** 位ΔΣ类型 **A/D** 转换器中使用的寄存器）

图 3-17. 数据存储空间与寻址方式的对应关系（μPD78F0444，F0454）

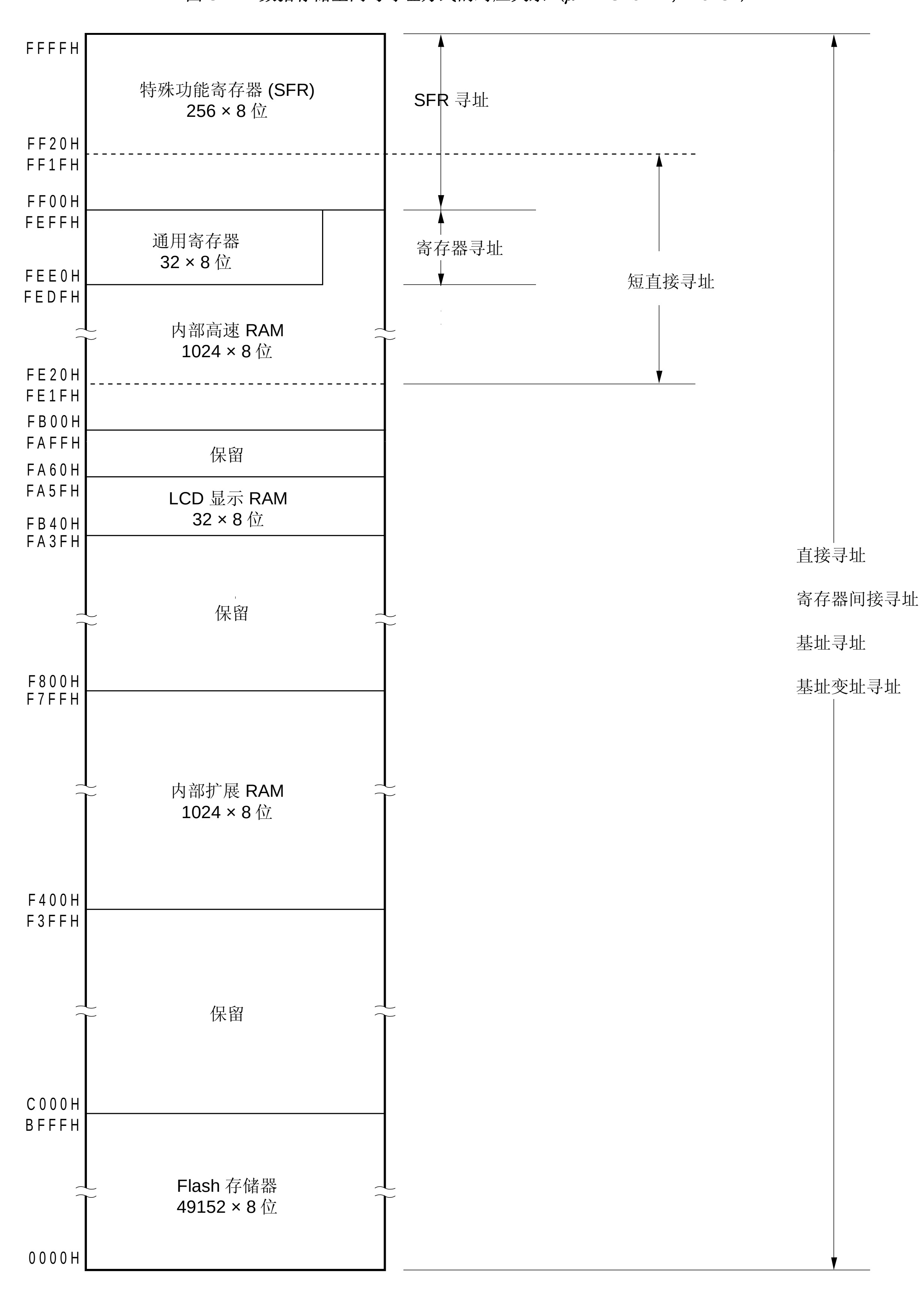
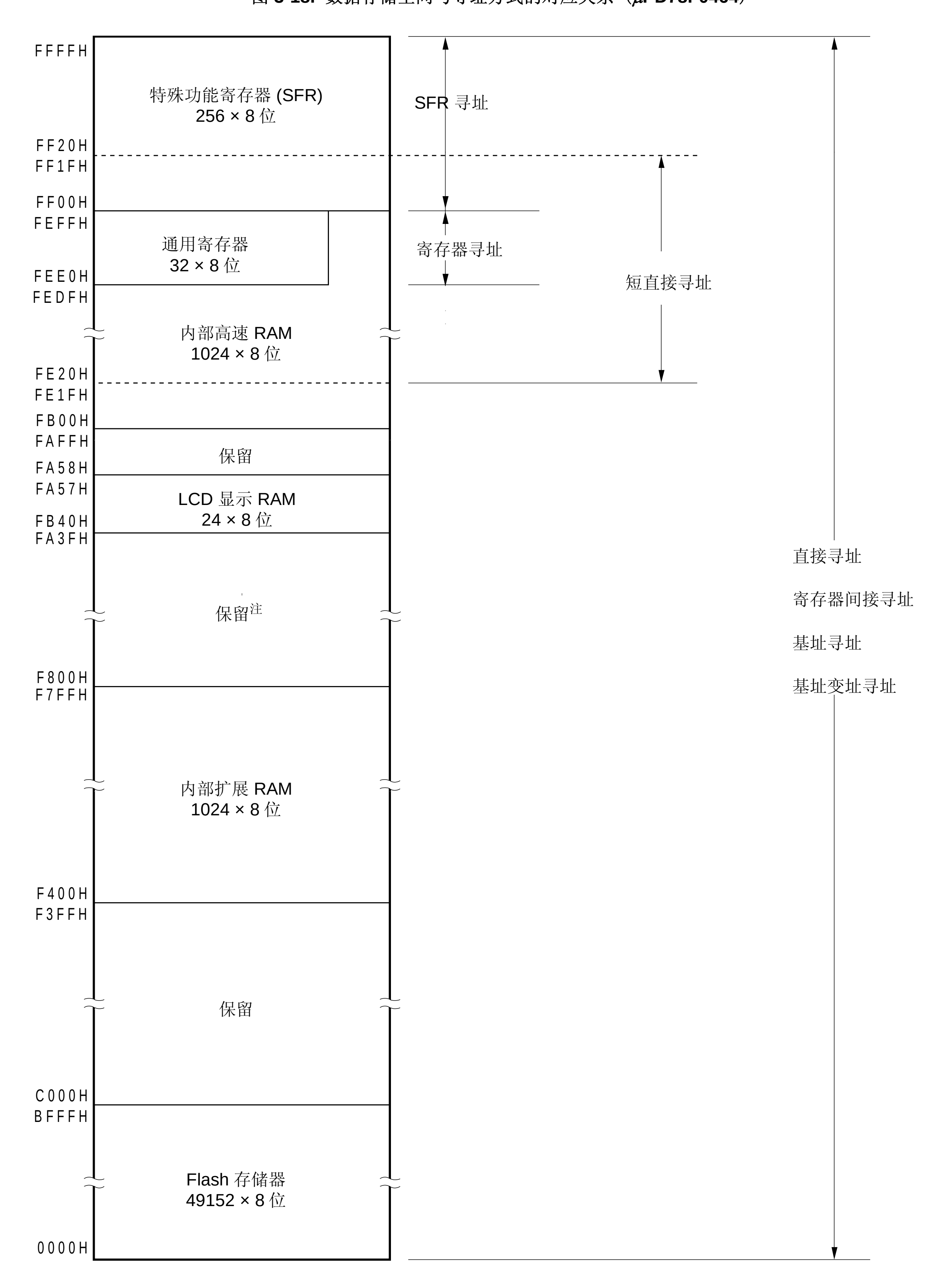


图 3-18. 数据存储空间与寻址方式的对应关系（μPD78F0464）



<R> 注 但是，可以使用 FA26H 和 FA27H（参见 **13.3 16 位ΔΣ类型 A/D 转换器中使用的寄存器**）

图 3-19. 数据存储空间与寻址方式的对应关系（μPD78F0445，F0455）

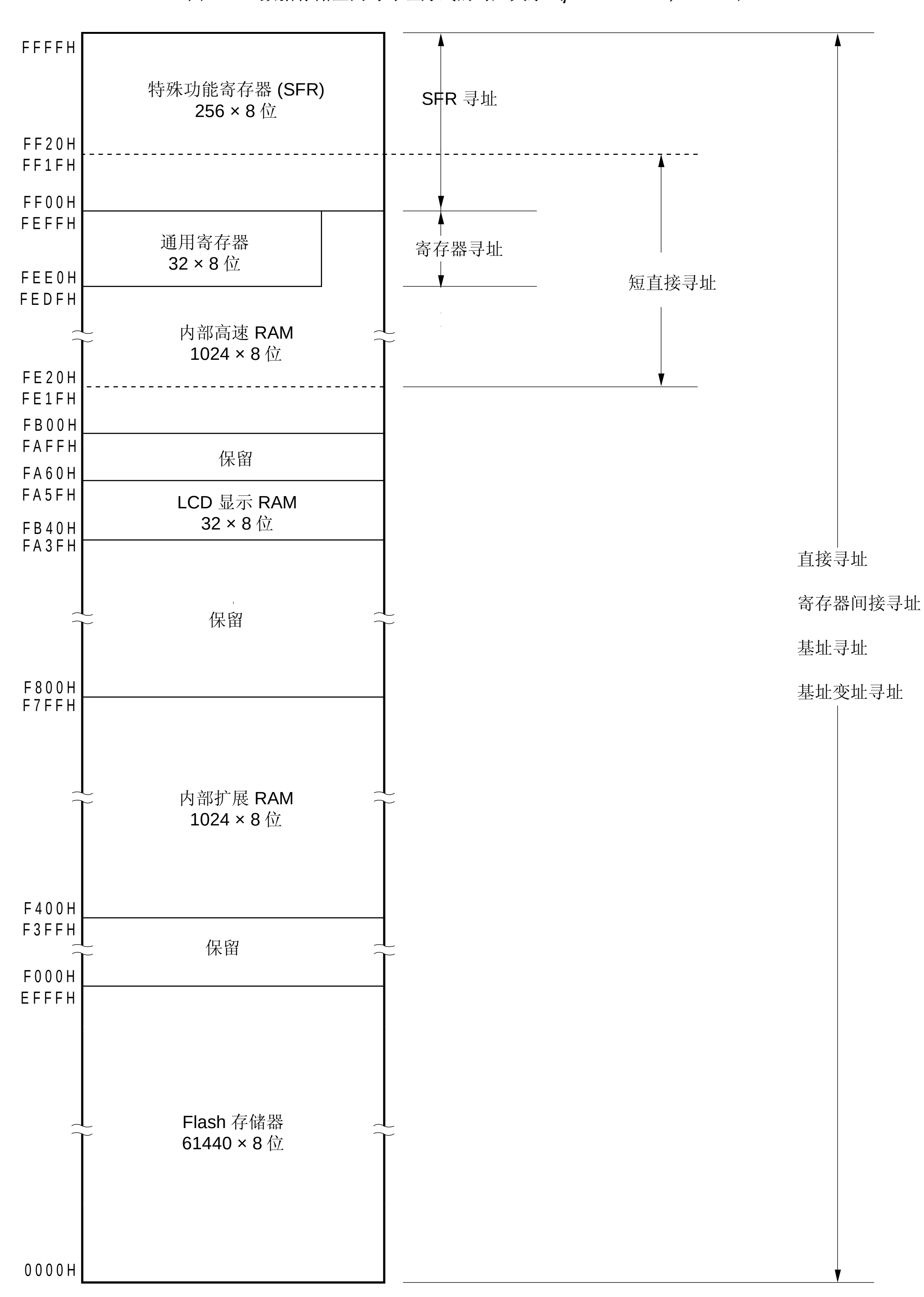
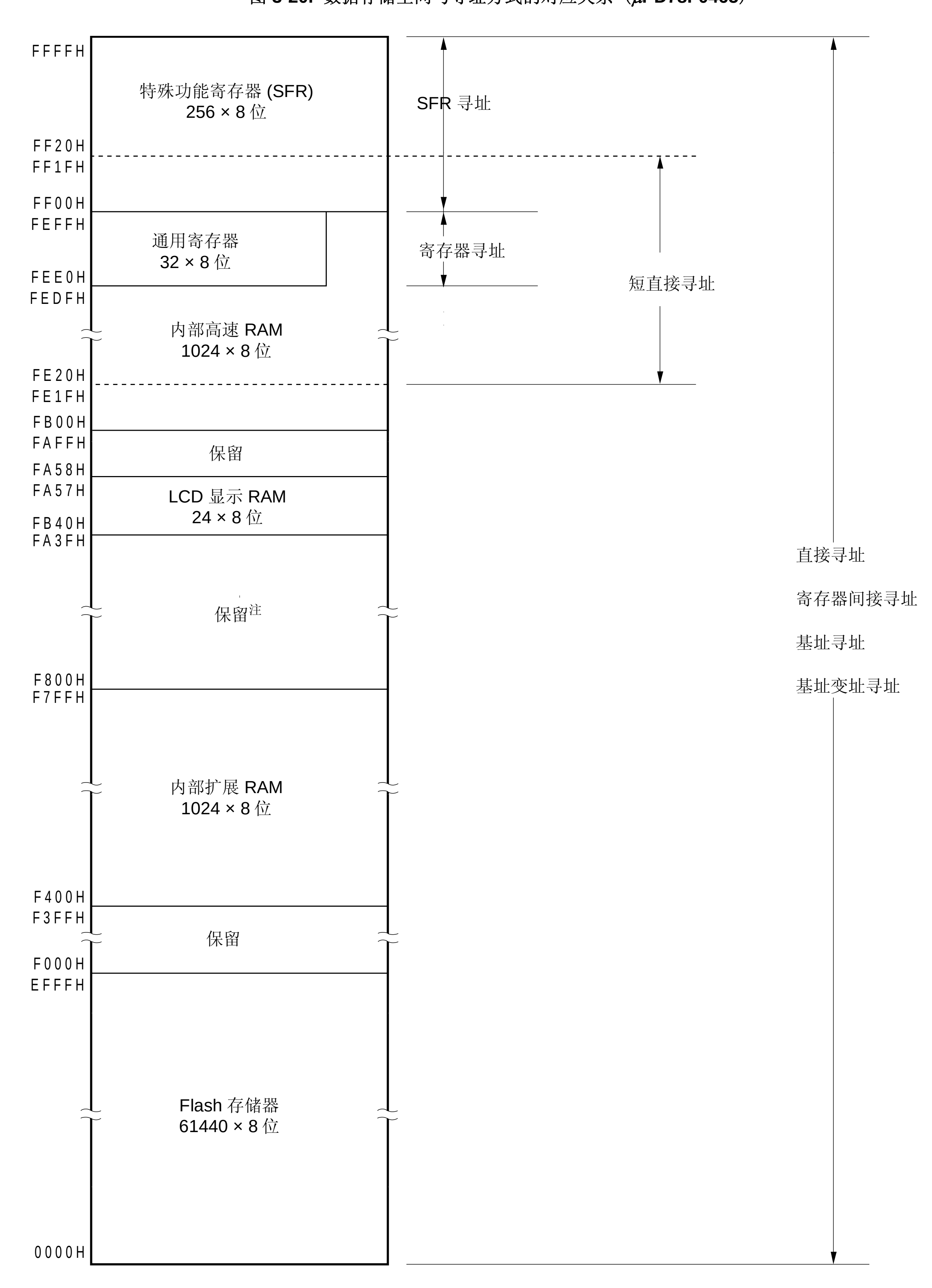


图 **3-20.** 数据存储空间与寻址方式的对应关系（**μPD78F0465**）



<R> 注 但是，可以使用 FA26H 和 FA27H（参见 **13.3 16** 位ΔΣ类型 **A/D** 转换器中使用的寄存器）

3.2 处理器寄存器

78K0/LE3 产品包括以下几种处理器寄存器。

3.2.1 控制寄存器

控制寄存器用于控制程序执行的顺序、状态和堆栈存储空间。程序计数器（PC）、程序状态字（PSW）和堆栈指针寄存器（SP）都属于控制寄存器。

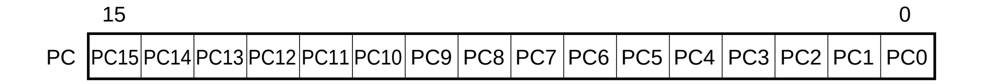
（1） 程序计数器（PC）

程序计数器是 16 位寄存器，用于存放下一条将要执行的指令的地址。

在正常操作时，根据获取的指令字节数量，程序计数器（PC）的值会自动累加。当执行分支指令时，则设置立即数据和寄存器内容。

复位信号的产生会将复位向量表中 0000H 和 0001H 地址上的值赋给程序计数器。

图 3-21 程序计数器的格式



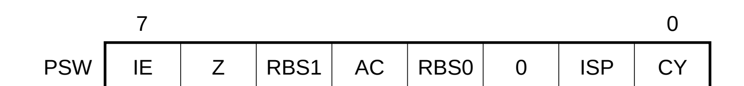
（2） 程序状态字（PSW）

程序状态字（PSW）是 8 位寄存器，由各种标志位组成，指令的执行会对其进行置位/重置。

响应中断请求时或执行 PUSH PSW 指令时，程序状态字的内容自动存入堆栈；在执行 RETB，RETI 和 POP PSW 指令时，程序状态字的值自动恢复。

复位信号的产生会将程序状态字的内容设置为 02H。

图 3-22 程序状态字的格式



（a） 中断允许标志（IE）

该标志用于控制 CPU 响应中断请求的操作。

当 IE 为 0 时，IE 标志被设置为中断禁止（DI）状态，所有可屏蔽中断的请求都被禁止。

当 IE 为 1 时，IE 标志被设置为中断使能（EI）状态，中断请求的响应由服务中的优先级标志（ISP）、各种中断源的中断屏蔽标志以及优先级指定标志来控制。

当执行 DI 指令或中断请求得到响应时，该标志被重置（0）；当执行 EI 指令时，该标志被置位（1）。

（b） 零标志 （Z）

当操作结果为 0 时，该标志被置位（1），其他情况下被重置（0）。

（c） 寄存器组选择标志（RBS0 和 RBS1）

这两位标志用于选择四组寄存器中的其中一组。

这两位标志位中存储的信息表示通过执行 SEL RBn 指令选择的寄存器组。

（d） 辅助进位标志（AC）

如果操作结果中第 3 位有进位或第 3 位有借位，该标志被置位（1），其他情况下被重置（0）。

（e） 服务中的优先级标志（ISP）

该标志用于管理可屏蔽向量中断中可以响应的优先级。当 ISP 为 0 时，由优先级指定标志寄存器（PR0L，PR0H，PR1L，PR1H）（参见 **21.3 （3） 优先级指定标志寄存器（PR0L，PR0H，PR1L，PR1H）**）指定为低优先级的向量中断请求无法响应。对请求的实际响应是由中断允许标志（IE）来控制。

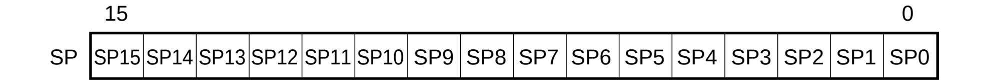
（f） 进位标志（CY）

该标志存储的是在执行加/减指令时出现的溢出或下溢。它也存储执行循环指令时移出的值，还可以在执行位操作指令时作为位累加器使用。

（3） 堆栈指针（SP）

这是 16 位的寄存器，用于存放存储器堆栈区域的起始地址。只有内部高速 RAM 区域可以被设置为堆栈区域。

图 3-23 堆栈指针的格式



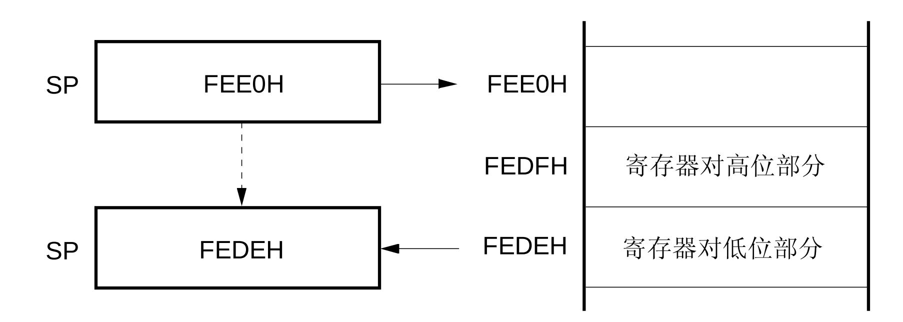
向堆栈写入（存入）数据之前，堆栈指针 SP 递减，而从堆栈中读出（恢复）数据之后，堆栈指针累加。

堆栈的数据存储/恢复操作过程如图 3-24 和 3-25 所示。

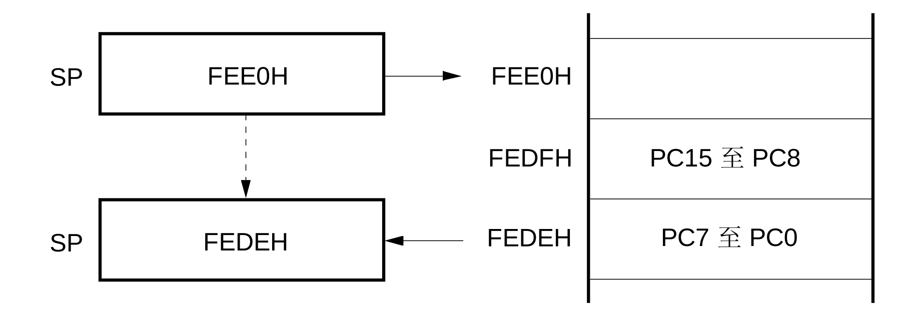
注意事项：由于复位信号产生时，SP 的内容不确定，所以在使用堆栈前请确保对 SP 进行初始化。

图 **3-24.** 将数据存入堆栈存储区

（**a**） **PUSH rp** 指令（当 **SP = FEE0H**）



（**b**） **CALL**，**CALLF**，**CALLT** 指令（当 **SP = FEE0H**）



（**c**） **Interrupt**，**BRK** 指令（当 **SP = FEE0H**）

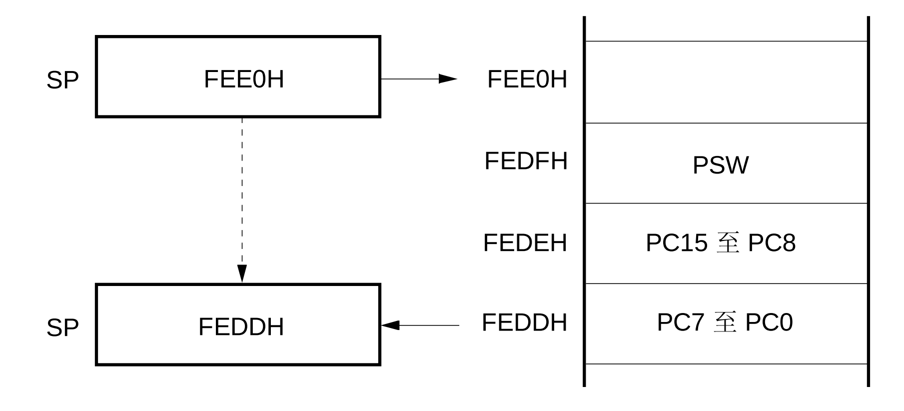
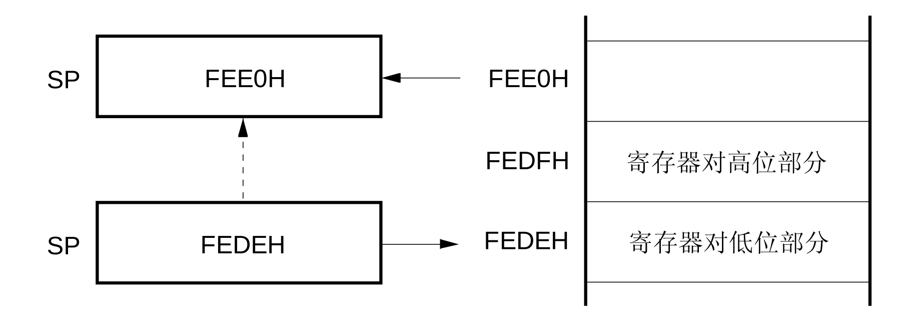
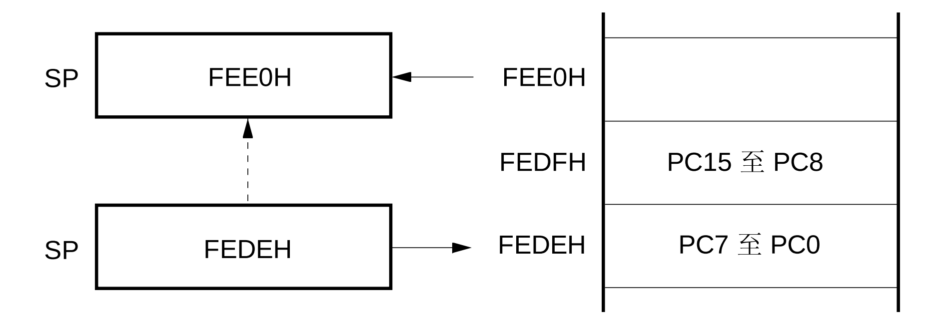


图 **3-25.** 从堆栈存储区恢复数据

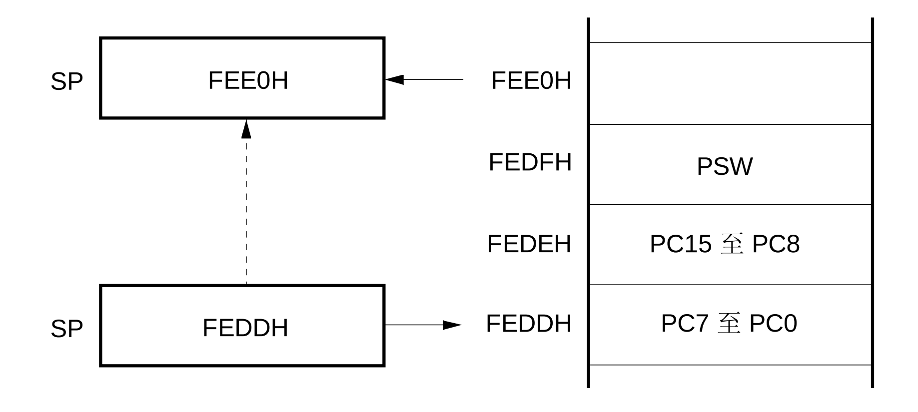
（**a**） **POP rp** 指令（当 **SP = FEDEH**）



（**b**） **RET** 指令（当 **SP = FEDEH**）



（**c**） **RETI**，**RETB** 指令（当 **SP = FEDDH**）



3.2.2 通用寄存器

通用寄存器映射到数据存储器特定地址（FEE0H 至 FEFFH）。通用寄存器共有四组，每组由 8 个 8 位寄存器（X，A，C，B，E，D，L 和 H）组成。

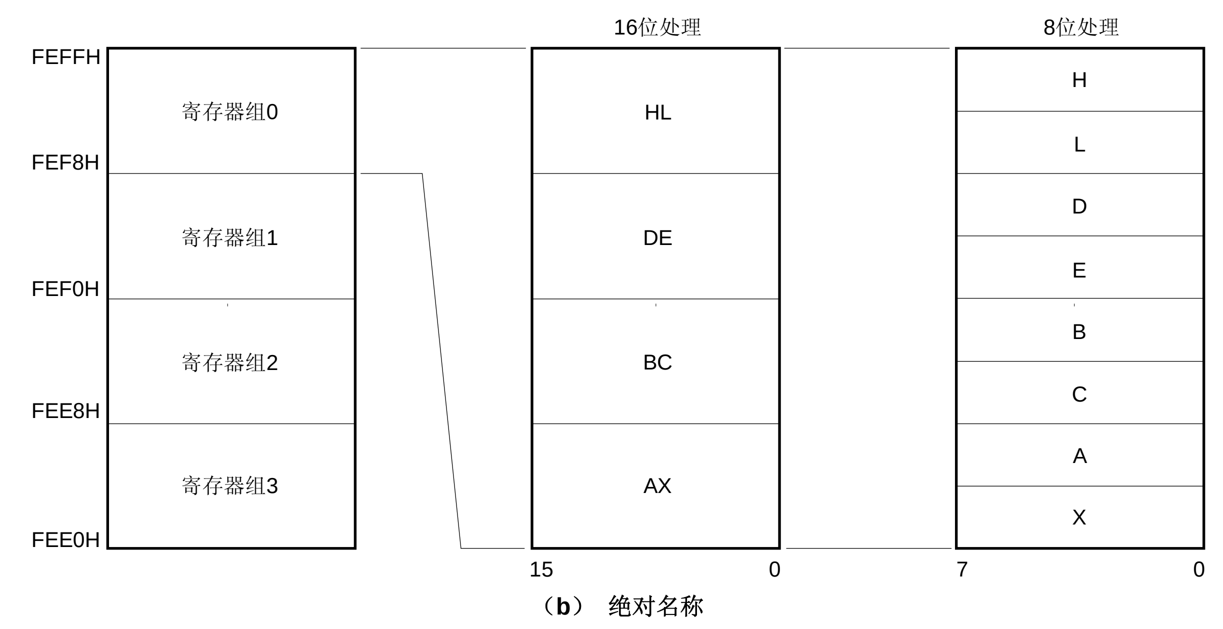
此外每个寄存器可作为一个 8 位寄存器使用，两个 8 位寄存器可以成对作为一个 16 位寄存器（AX，BC，DE 和 HL）使用。

描述通用寄存器时，可以按照功能名称（X，A，C，B，E，D，L，H，AX，BC，DE 和 HL）或绝对名称（R0 至 R7，RP0 至 RP3）。

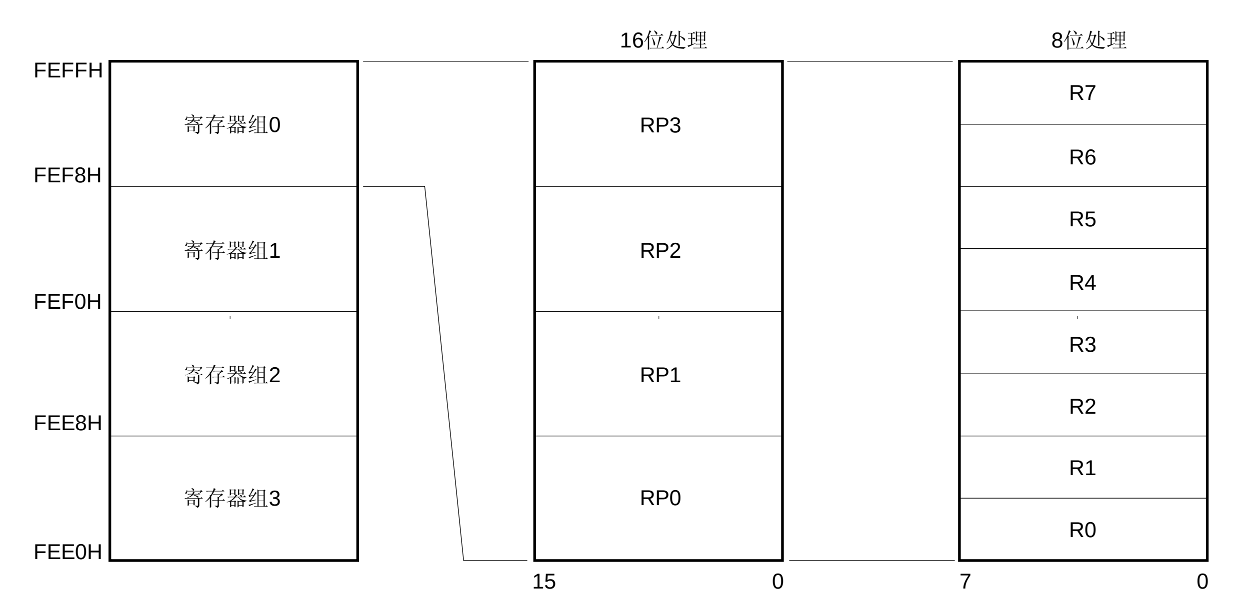
指令执行所需的寄存器组由 CPU 控制指令（SEL RBn）来设置。因为这种 4 个寄存器组的结构配置，通过一个用于正常处理的寄存器和另一个用于中断处理的寄存器之间的切换，可以创建高效率的程序。

图 3-26 通用寄存器结构

（a） 功能名称



（b） 绝对名称



3.2.3 特殊功能寄存器 （SFR）

与通用寄存器不同，每个特殊功能寄存器都有特定的功能。

SFR 分配在 CPU 的 FF00H 至 FFFFH 区域和 LCD 控制器/驱动器的 LCDCTL 的 00H 到 03H 区域。

特殊功能寄存器可以像通用寄存器那样，使用运算指令、传送指令以及位操作指令进行操作。根据特殊功能寄存器的类型不同，可操作的单元也不同，可以按照1位、8位和16位单元。

每种单元操作的描述如下。

- 1 位操作

  描述 1 位操作指令的操作数（sfr.bit）时，应该使用汇编程序的保留符号。

  该操作也可由一个地址来定义。

- 8 位操作

  描述 8 位操作指令的操作数（sfr）时，应该使用汇编程序的保留符号。

  该操作也可由一个地址来定义。

- 16 位操作

  描述 16 位操作指令的操作数（sfrp）时，应该使用汇编程序的保留符号。

  指定地址时，应该使用偶地址来描述。

表 3-8 是特殊功能寄存器列表。表中各项术语的含义如下。

- 符号

  符号表示特殊功能寄存器的地址，在 RA78K0 中该符号是保留字，在 CC78K0 中使用#pragma sfr 指令将其定义为一个 sfr 变量。在使用 RA78K0，ID78K0-QB 和 SM+时，符号可以被作为指令德操作数来写入。

- R/W

  表示特殊功能寄存器可读或可写。

  R/W： 可读/可写

  R： 只读

  W： 只写

- 可操作的位单元

  表示可操作的位单元 （1，8，或 16 位）。"–" 表示不可操作的位单元 。

- 复位后

  表示在产生复位信号时，每个寄存器的状态

表 3-8 特殊功能寄存器列表（1/5）

地址	特殊功能寄存器（SFR）名称		符号	R/W	可操作位单元			复位后
					1 位	8 位	16 位	
FF00H	接收缓冲寄存器 6		RXB6	R	–	√	–	FFH
FF01H	端口寄存器 1		P1	R/W	√	√	–	00H
FF02H	端口寄存器 2		P2	R/W	√	√	–	00H
FF03H	端口寄存器 3		P3	R/W	√	√	–	00H
FF04H	端口寄存器 4		P4	R/W	√	√	–	00H
FF05H	发送缓冲寄存器 6		TXB6	R/W	–	√	–	FFH
FF06H	10 位 A/D 转换结果寄存器[注]		ADCR	R	–	–	√	0000H
FF07H		8 位 A/D 转换结果寄存器 H[注]	ADCRH	R	–	√	–	00H
FF08H	端口寄存器 8		P8	R/W	√	√	–	00H
FF09H	端口寄存器 9		P9	R/W	√	√	–	00H
FF0AH	端口寄存器 10		P10	R/W	√	√	–	00H
FF0BH	端口寄存器 11		P11	R/W	√	√	–	00H
FF0CH	端口寄存器 12		P12	R/W	√	√	–	00H
FF0EH	端口寄存器 14		P14	R/W	√	√	–	00H
FF0FH	端口寄存器 15		P15	R/W	√	√	–	00H
FF10H	16 位定时器计数器 00		TM00	R	–	–	√	0000H
FF11H								
FF12H	16 位定时器捕获/比较寄存器 000		CR000	R/W	–	–	√	0000H
FF13H								
FF14H	16 位定时器捕获/比较寄存器 010		CR010	R/W	–	–	√	0000H
FF15H								
FF16H	8 位定时器计数器 50		TM50	R	–	√	–	00H
FF17H	8 位定时器比较寄存器 50		CR50	R/W	–	√	–	00H
FF18H	8 位定时器 H 比较寄存器 00		CMP00	R/W	–	√	–	00H
FF19H	8 位定时器 H 比较寄存器 10		CMP10	R/W	–	√	–	00H
FF1AH	8 位定时器 H 比较寄存器 01		CMP01	R/W	–	√	–	00H
FF1BH	8 位定时器 H 比较寄存器 11		CMP11	R/W	–	√	–	00H
FF1FH	串行 I/O 移位寄存器 10		SIO10	R	–	√	–	00H
FF20H	端口功能寄存器 1		PF1	R/W	√	√	–	00H
FF21H	端口模式寄存器 1		PM1	R/W	√	√	–	FFH
FF22H	端口模式寄存器 2		PM2	R/W	√	√	–	FFH
FF23H	端口模式寄存器 3		PM3	R/W	√	√	–	FFH
FF24H	端口模式寄存器 4		PM4	R/W	√	√	–	FFH
FF28H	端口模式寄存器 8		PM8	R/W	√	√	–	FFH
FF29H	端口模式寄存器 9		PM9	R/W	√	√	–	FFH
FF2AH	端口模式寄存器 10		PM10	R/W	√	√	–	FFH
FF2BH	端口模式寄存器 11		PM11	R/W	√	√	–	FFH
FF2CH	端口模式寄存器 12		PM12	R/W	√	√	–	FFH
FF2EH	端口模式寄存器 14		PM14	R/W	√	√	–	FFH
FF2FH	端口模式寄存器 15		PM15	R/W	√	√	–	FFH

注　仅限μPD78F045x 和 78F046x。

表 3-8 特殊功能寄存器列表（2/5）

地址	特殊功能寄存器（SFR）名称	符号	R/W	可操作位单元			复位后
				1 位	8 位	16 位	
FF30H	内部高速振荡器微调寄存器	HIOTRM	R/W	–	√	–	10H
FF31H	上拉电阻选择寄存器 1	PU1	R/W	√	√	–	00H
FF33H	上拉电阻选择寄存器 3	PU3	R/W	√	√	–	00H
FF34H	上拉电阻选择寄存器 4	PU4	R/W	√	√	–	00H
FF38H	上拉电阻选择寄存器 8	PU8	R/W	√	√	–	00H
FF39H	上拉电阻选择寄存器 9	PU9	R/W	√	√	–	00H
FF3AH	上拉电阻选择寄存器 10	PU10	R/W	√	√	–	00H
FF3BH	上拉电阻选择寄存器 11	PU11	R/W	√	√	–	00H
FF3CH	上拉电阻选择寄存器 12	PU12	R/W	√	√	–	00H
FF3DH	上拉电阻选择寄存器 13	PU13	R/W	√	√	–	00H
FF3EH	上拉电阻选择寄存器 14	PU14	R/W	√	√	–	00H
FF3FH	上拉电阻选择寄存器 15	PU15	R/W	√	√	–	00H
FF40H	时钟输出选择寄存器	CKS	R/W	√	√	–	00H
FF41H	8 位定时器比较寄存器 51	CR51	R/W	–	√	–	00H
FF42H	8 位定时器 H 模式寄存器 2	TMHMD2	R/W	√	√	–	00H
FF43H	8 位定时器模式控制寄存器 51	TMC51	R/W	√	√	–	00H
FF44H	8 位定时器 H 比较寄存器 02	CMP02	R/W	–	√	–	00H
FF45H	8 位定时器 H 比较寄存器 12	CMP12	R/W	–	√	–	00H
FF47H	MCG 状态寄存器	MC0STR	R	√	√	–	00H
FF48H	外部中断上升沿使能寄存器	EGP	R/W	√	√	–	00H
FF49H	外部中断下降沿使能寄存器	EGN	R/W	√	√	–	00H
FF4AH	MCG 发送缓冲寄存器	MC0TX	R/W	–	√	–	FFH
FF4BH	MCG 发送位计数规范寄存器	MC0BIT	R/W	–	√	–	07H
FF4CH	MCG 控制寄存器 0	MC0CTL0	R/W	√	√	–	10H
FF4DH	MCG 控制寄存器 1	MC0CTL1	R/W	–	√	–	00H
FF4EH	MCG 控制寄存器 2	MC0CTL2	R/W	–	√	–	1FH
FF4FH	输入切换控制寄存器	ISC	R/W	√	√	–	00H
FF50H	异步串行接口操作模式寄存器 6	ASIM6	R/W	√	√	–	01H
FF51H	8 位定时器计数器 52	TM52	R	–	√	–	00H
FF53H	异步串行接口接收错误状态寄存器 6	ASIS6	R	–	√	–	00H
FF54H	实时计数器时钟选择寄存器	RTCCL	R/W	√	√	–	00H
FF55H	异步串行接口传送状态寄存器 6	ASIF6	R	–	√	–	00H
FF56H	时钟选择寄存器 6	CKSR6	R/W	–	√	–	00H
FF57H	波特率发生器控制寄存器 6	BRGC6	R/W	–	√	–	FFH
FF58H	异步串行接口控制寄存器 6	ASICL6	R/W	√	√	–	16H
FF59H	8 位定时器比较寄存器 52	CR52	R/W	–	√	–	00H
FF5BH	定时器时钟选择寄存器 52	TCL52	R/W	√	√	–	00H
FF5CH	8 位定时器模式控制寄存器 52	TMC52	R/W	√	√	–	00H

表 **3-8** 特殊功能寄存器列表（**3/5**）

地址	特殊功能寄存器（SFR）名称	符号	R/W	可操作位单元			复位后
				1 位	8 位	16 位	
FF60H FF61H	副计数寄存器	RSUBC	R	–	–	√	0000H
FF62H	秒计数寄存器	SEC	R/W	–	√	–	00H
FF63H	分钟计数寄存器	MIN	R/W	–	√	–	00H
FF64H	小时计数寄存器	HOUR	R/W	–	√	–	12H
FF65H	星期计数寄存器	WEEK	R/W	–	√	–	00H
FF66H	日计数寄存器	DAY	R/W	–	√	–	01H
FF67H	月计数寄存器	MONTH	R/W	–	√	–	01H
FF68H	年计数寄存器	YEAR	R/W	–	√	–	00H
FF69H	8 位定时器 H 模式寄存器 0	TMHMD0	R/W	√	√	–	00H
FF6AH	定时器时钟选择寄存器 50	TCL50	R/W	√	√	–	00H
FF6BH	8 位定时器模式控制寄存器 50	TMC50	R/W	√	√	–	00H
FF6CH	8 位定时器 H 模式寄存器 1	TMHMD1	R/W	√	√	–	00H
FF6DH	8 位定时器 H 载波控制寄存器 1	TMCYC1	R/W	√	√	–	00H
FF6EH	按键返回模式寄存器	KRM	R/W	√	√	–	00H
FF6FH	8 位定时器计数器 51	TM51	R	–	√	–	00H
FF70H	异步串行接口操作模式寄存器 0	ASIM0	R/W	√	√	–	01H
FF71H	波特率发生器控制寄存器 0	BRGC0	R/W	–	√	–	1FH
FF72H	接收缓冲寄存器 0	RXB0	R	–	√	–	FFH
FF73H	异步串行接口接收错误状态寄存器 0	ASIS0	R	–	√	–	00H
FF74H	发送移位寄存器 0	TXS0	W	–	√	–	FFH
FF75H	16 位 ΔΣ 型 A/D 转换结束通道寄存器注 1	ADDSTR	R	–	√	–	00H
FF7CH	ΔΣ A/D 转换器控制寄存器 0 注 1	ADDCTL0	R/W	√	√	–	00H
FF7DH	ΔΣ A/D 转换器控制寄存器 1 注 1	ADDCTL1	R/W	√	√	–	00H
FF7EH	16 位 ΔΣ A/D 转换结果寄存器注 1	ADDCR	R	–	–	√	0000H
FF7FH	8 位 ΔΣ A/D 转换结果寄存器注 1	ADDCRH	R	–	√	–	00H
FF80H	串行操作模式寄存器 10	CSIM10	R/W	√	√	–	00H
FF81H	串行时钟选择寄存器 10	CSIC10	R/W	√	√	–	00H
FF82H	钟表错误修正寄存器	SUBCUD	R/W	√	√	–	00H
FF84H	发送缓冲寄存器 10	SOTB10	R/W	–	√	–	00H
FF86H	闹钟分钟寄存器	ALARMWM	R/W	–	√	–	00H
FF87H	闹钟小时寄存器	ALARMWH	R/W	–	√	–	12H
FF88H	闹钟星期寄存器	ALARMWW	R/W	–	√	–	00H
FF89H	实时计数器控制寄存器 0	RTCC0	R/W	√	√	–	00H
FF8AH	实时计数器控制寄存器 1	RTCC1	R/W	√	√	–	00H
FF8BH	实时计数器控制寄存器 2	RTCC2	R/W	√	√	–	00H
FF8CH	定时器时钟选择寄存器 51	TCL51	R/W	√	√	–	00H
FF8DH	A/D 转换器模式寄存器注 2	ADM	R/W	√	√	–	00H
FF8EH	模拟输入通道规范寄存器注 2	ADS	R/W	√	√	–	00H
FF8FH	A/D 端口配置寄存器 0 注 2	ADPC0	R/W	√	√	–	08H

注 **1.** 仅限μPD78F046x。

2. 仅限μPD78F045x 和 78F046x。

表 3-8 特殊功能寄存器列表（4/5）

地址	特殊功能寄存器（SFR）名称	符号	R/W	可操作位单元			复位后
FF90H	串行操作模式规范寄存器 0	CSIMA0	R/W	√	√	–	00H
FF91H	串行状态寄存器 0	CSIS0	R/W	√	√	–	00H
FF92H	串行触发寄存器 0	CSIT0	R/W	√	√	–	00H
FF93H	分频值选择寄存器 0	BRGCA0	R/W	–	√	–	03H
FF94H	自动数据传输地址指针规范寄存器 0	ADTP0	R/W	–	√	–	00H
FF95H	自动数据传输间隔规范寄存器	ADTI0	R/W	–	√	–	00H
FF96H	串行 I/O 移位寄存器 0	SIOA0	R/W	–	√	–	00H
FF97H	自动数据传输地址计数寄存器 0	ADTC0	R	–	√	–	00H
FF99H	看门狗定时器使能寄存器	WDTE	R/W	–	√	–	注 1 1AH/9AH
FF9AH	遥控接收控制寄存器	RMCN	R/W	√	√	–	00H
FF9BH	遥控接收数据寄存器	RMDR	R	–	√	–	00H
FF9CH	遥控移位寄存器接收计数器寄存器	RMSCR	R	–	√	–	00H
FF9FH	时钟操作模式选择寄存器	OSCCTL	R/W	√	√	–	00H
FFA0H	内部振荡模式寄存器	RCM	R/W	√	√	–	80H 注 2
FFA1H	主时钟模式寄存器	MCM	R/W	√	√	–	00H
FFA2H	主 OSC 控制寄存器	MOC	R/W	√	√	–	80H
FFA3H	振荡稳定时间计数器状态寄存器	OSTC	R	√	√	–	00H
FFA4H	振荡稳定时间选择寄存器	OSTS	R/W	–	√	–	05H
FFA5H	遥控接收 GPHS 比较寄存器	RMGPHS	R/W	–	√	–	00H
FFA6H	遥控接收 GPHL 比较寄存器	RMGPHL	R/W	–	√	–	00H
FFA7H	遥控接收 DLS 比较寄存器	RMDLS	R/W	–	√	–	00H
FFA8H	遥控接收 DLL 比较寄存器	RMDLL	R/W	–	√	–	00H
FFA9H	遥控接收 DH0S 比较寄存器	RMDH0S	R/W	–	√	–	00H
FFAAH	遥控接收 DH0L 比较寄存器	RMDH0L	R/W	–	√	–	00H
FFABH	遥控接收移位寄存器	RMSR	R	–	√	–	00H
FFACH	复位控制标志寄存器	RESF	R	–	√	–	00H 注 3
FFADH	遥控接收 DH1S 比较寄存器	RMDH1S	R/W	–	√	–	00H
FFAEH	遥控接收 DH1L 比较寄存器	RMDH1L	R/W	–	√	–	00H
FFAFH	遥控接收结束宽度选择寄存器	RMER	R/W	–	√	–	00H

注 **1.** WDTE 的复位值由选项字节的设置决定。

2. 该寄存器的值在复位释放后立即为 00H， 但当高速内部振荡器振荡稳定之后，自动变为 80H 。

3. RESF 的复位值根据复位源而不同。

表 **3-8** 特殊功能寄存器列表（**5/5**）

地址	特殊功能寄存器（SFR）名称	符号		R/W	可操作位单元			复位后
					1 位	8 位	16 位	
FFB0H	LCD 模式寄存器	LCDMD		R/W	√	√	–	00H
FFB1H	LCD 显示模式寄存器	LCDM		R/W	√	√	–	00H
FFB2H	LCD 时钟控制寄存器 0	LCDC0		R/W	√	√	–	00H
FFB5H	端口功能寄存器 2[注1]	PF2		R/W	√	√	–	00H
FFB6H	端口功能寄存器 ALL	PFALL		R/W	√	√	–	00H
FFBAH	16 位定时器模式控制寄存器 00	TMC00		R/W	√	√	–	00H
FFBBH	预分频模式寄存器 00	PRM00		R/W	√	√	–	00H
FFBCH	捕捉/比较控制寄存器 00	CRC00		R/W	√	√	–	00H
FFBDH	16 位定时器输出控制寄存器 00	至 C00		R/W	√	√	–	00H
FFBEH	低电压检测寄存器	LVIM		R/W	√	√	–	00H[注2]
FFBFH	低电压检测电平选择寄存器	LVIS		R/W	√	√	–	00H[注2]
FFE0H	中断请求标志寄存器 0L	IF0	IF0L	R/W	√	√	√	00H
FFE1H	中断请求标志寄存器 0H		IF0H	R/W	√	√		00H
FFE2H	中断请求标志寄存器 1L	IF1	IF1L	R/W	√	√	√	00H
FFE3H	中断请求标志寄存器 1H		IF1H	R/W	√	√		00H
FFE4H	中断屏蔽标志寄存器 0L	MK0	MK0L	R/W	√	√	√	FFH
FFE5H	中断屏蔽标志寄存器 0H		MK0H	R/W	√	√		FFH
FFE6H	中断屏蔽标志寄存器 1L	MK1	MK1L	R/W	√	√	√	FFH
FFE7H	中断屏蔽标志寄存器 1H		MK1H	R/W	√	√		FFH
FFE8H	优先级指定标志寄存器 0L	PR0	PR0L	R/W	√	√	√	FFH
FFE9H	优先级指定标志寄存器 0H		PR0H	R/W	√	√		FFH
FFEAH	优先级指定标志寄存器 1L	PR1	PR1L	R/W	√	√	√	FFH
FFEBH	优先级指定标志寄存器 1H		PR1H	R/W	√	√		FFH
FFF0H	内部存储器容量切换寄存器[注3]	IMS		R/W	–	√	–	CFH
FFF4H	内部扩展 RAM 容量切换寄存器[注3]	IXS		R/W	–	√	–	0CH
FFF9H	遥控接收中断状态寄存器	INTS		R	√	√	–	00H
FFFAH	遥控接收中断状态清零寄存器	INTC		R/W	√	√	–	00H
FFFBH	处理器时钟控制寄存器	PCC		R/W	√	√	–	01H

注 **1.** 仅限 8F044x 和 78F045x 。

2. LVIM 和 LVIS 的复位值根据复位源而不同。

3. 不论内部存储器容量的大小，所有 78K0/ LE3 产品的内部存储器容量切换寄存器（IMS）和内部扩展 RAM 容量切换寄存器（IXS）的初始值都是固定的（IMS = CFH，IXS = 0CH）。因此，请根据下表对每种产品进行相应设置。

Flash 存储器版本（78K0/LE3）	IMS	IXS	ROM 容量	内部高速 RAM 容量	内部扩展 RAM 容量
μPD78F0441，78F0451，78F0461	04H	0CH	16 KB	768 字节	–
μPD78F0442，78F0452，78F0462	C6H		24 KB	1 KB	
μPD78F0443，78F0453，78F0463	C8H		32 KB		
μPD78F0444，78F0454，78F0464	CCH	0AH	48 KB		1 KB
μPD78F0445，78F0455，78F0465	CFH		60 KB		

3.3 指令地址寻址

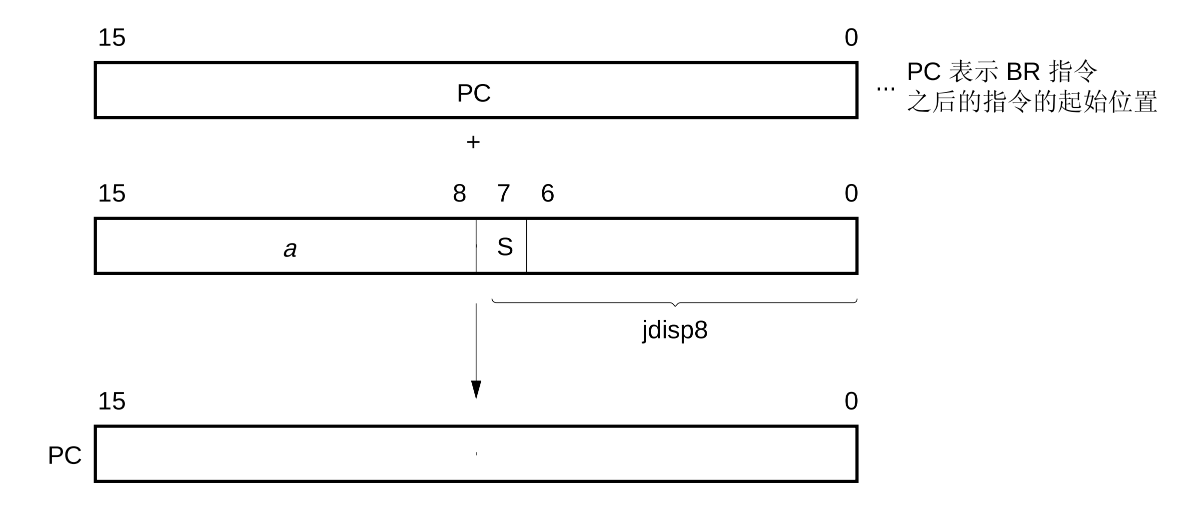
一条指令的地址是由程序计数器（PC）的内容决定。根据每次执行正常指令时将要获取的下一条指令的字节数量，程序计数器（PC）的内容自动增加（每个字节加 1）。执行转移指令时，将程序计数器（PC）的内容设置为转移目标信息，并按以下寻址方式跳转。（每条指令的详细信息，请参阅 **78K/0 系列指令用户手册（U12326E）**）。

3.3.1 相对寻址

[功能]

将一条指令代码的 8 位立即数（偏移量：jdisp8）与下一条指令的起始地址相加，结果赋给程序计数器（PC），然后跳转。这个偏移量是带符号数 2 的补码（–128 至+127），其中第 7 位是符号位。
换句话说，相对寻址中包含的跳转范围是从下一条指令起始地址的-128 到+127 之内。
当执行“BR $addr16”指令或条件转移指令时，将执行相对寻址功能。

[图示]



当 S = 0，a 的所有位是 0.
当 S = 1，a 的所有位是 1.

当S = 0时，α 的所有位均为0。
当 S = 1 时，α 的所有位均为 1。

3.3.2 立即寻址

[功能]

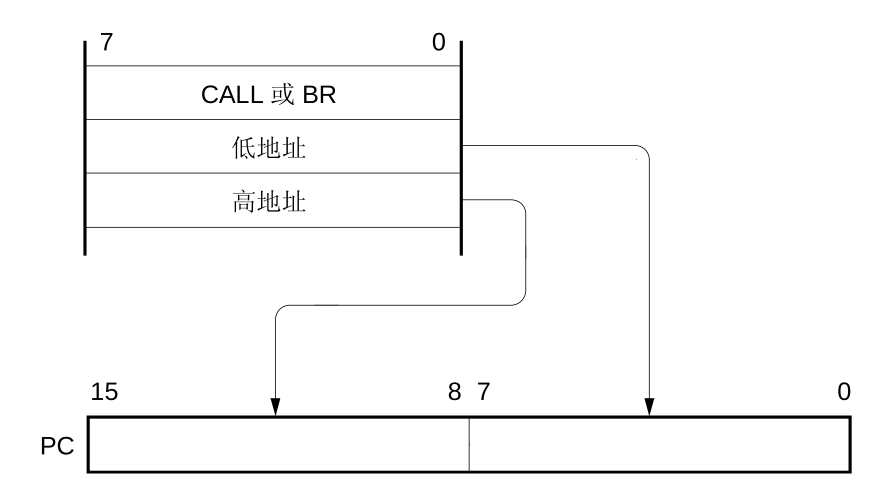
将指令字中的立即数赋给程序计数器（PC），然后跳转到该地址。

在执行“CALL !addr16”指令、 “BR !addr16”指令或 “CALLF !addr11” 指令时，将执行立即寻址功能。

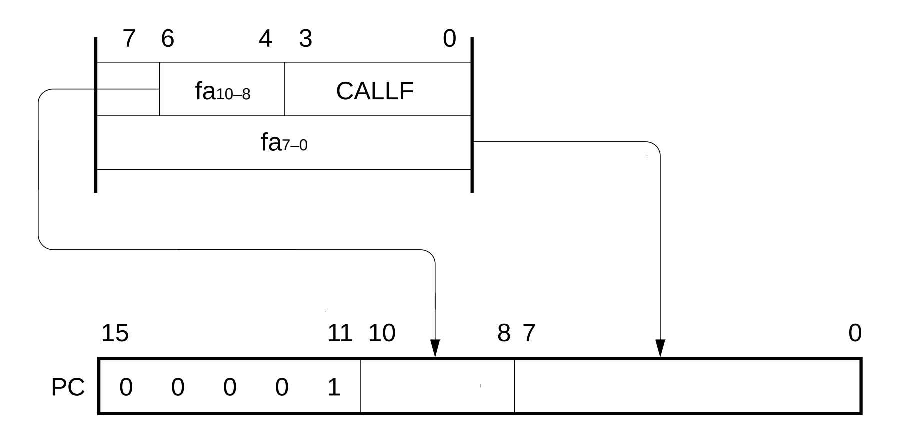
CALL !addr16 和 BR !addr16 指令的转移范围是整个存储器空间。CALLF !addr11 指令将会跳转到 0800H 至 0FFFH 区域。

[图示]

CALL !addr16 和 BR !addr16 指令的情况



CALLF !addr11 指令的情况



3.3.3 表间接寻址

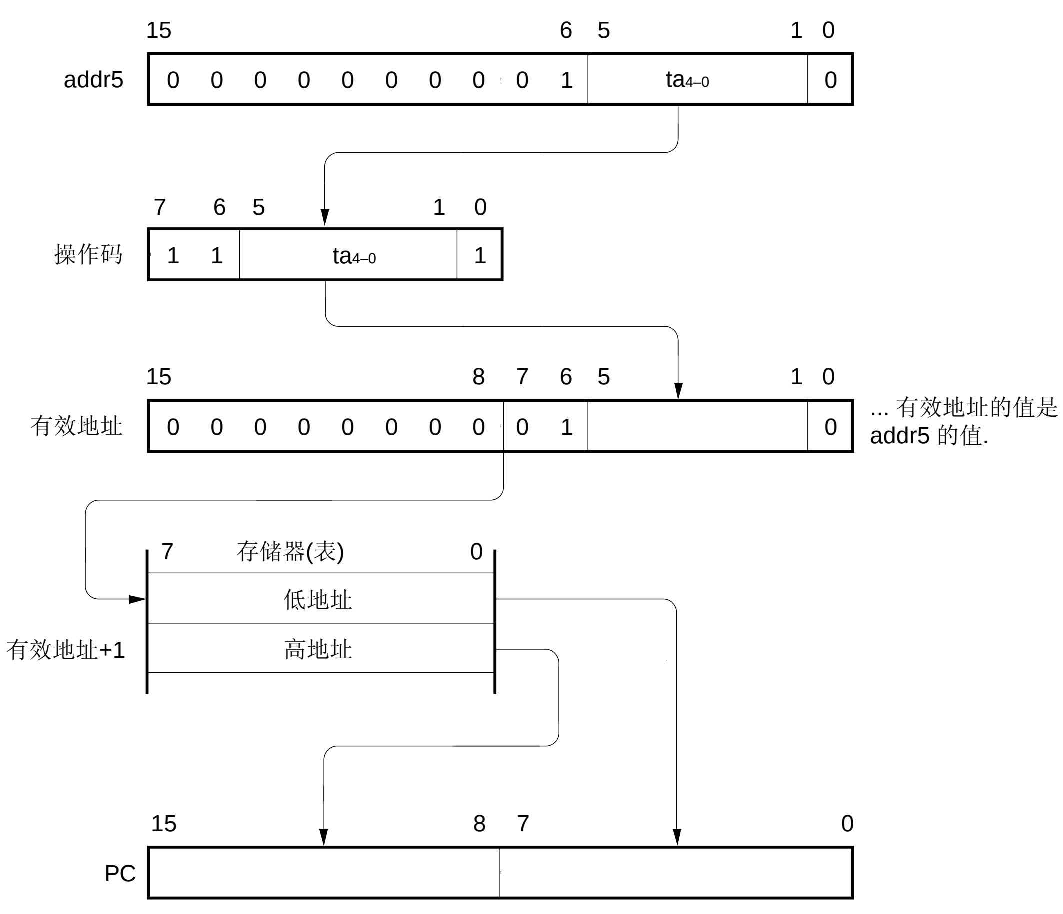
[功能]

通过指令码立即数的第 1 位到第 5 位的立即数，对特定位置的表的内容（跳转目的地址）寻址，并将其赋给程序计数器（PC），然后转向该地址执行程序。

在执行 CALLT [addr5]指令时，进行表间接寻址。

该指令参考存储在表 40H 至 7FH 中的地址，转移范围可以是整个存储器空间。

<R> **[图示]**



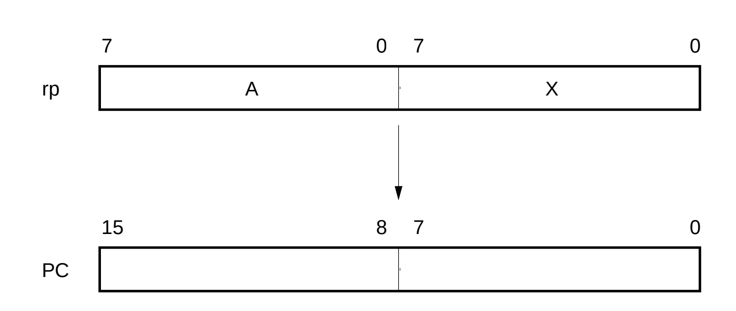
3.3.4 寄存器寻址

[功能]

通过指令字指定寄存器对（AX）的内容，并将其赋给程序计数器（PC），然后进行跳转。

当执行“BR AX”指令时，需要使用寄存器寻址功能。

[图示]



3.4 操作数地址寻址

以下方法用于指定指令执行期间需要操作的寄存器和存储器（寻址）。

3.4.1 隐含寻址

[功能]

该寄存器用作通用寄存器中的累加器（A 和 AX），可以自动（隐含）寻址。

在 78K0/LE3 系列指令字中，下列指令采用隐含寻址方式。

指令	隐含寻址所指定的寄存器
MULU	A 寄存器存放被乘数，AX 寄存器存放运算结果
DIVUW	AX 寄存器用于存放被除数和商
ADJBA/ADJBS	A 寄存器存放进行十进制调整后的数值
ROR4/ROL4	A 寄存器存放用于数字循环移位的数字数据

[操作数格式]

因为指令自动采用隐含寻址方式，所以无需特定的操作数格式。

[描述举例]

MULU X 的情况

这是 8 位乘 8 位的乘法运算指令，A 寄存器与 X 寄存器相乘的结果存放在 AX 中。在这个例子中 A 寄存器与 AX 寄存器均由隐含寻址方式指定。

3.4.2 寄存器寻址

[功能]

寄存器寻址方式将指定的通用寄存器作为操作数进行访问，并由寄存器组选择标志（RBS0 至 RBS1）和指令码中的寄存器标识码来指定需要访问的通用寄存器。

当具有下列操作数格式的指令执行时，采用寄存器寻址方式。如果使用 8 位寄存器，则指令码中有 3 位用于在八个寄存器中指定其中之一。

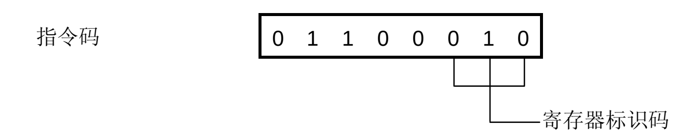
[操作数格式]

标识符	描述
r	X，A，C，B，E，D，L，H
rp	AX，BC，DE，HL

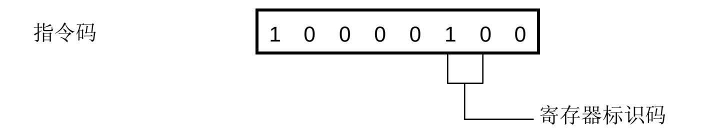
'r' 和 'rp'可用绝对名称（R0 至 R7 以及 RP0 至 RP3），也可以用功能名称（X，A，C，B，E，D，L，H，AX，BC，DE 以及 HL）来描述。

[描述举例]

MOV A，C;　当选择通用寄存器 C 为“r”时



INCW DE;　当选择通用寄存器组 DE 为“rp”时



3.4.3 直接寻址

[功能]

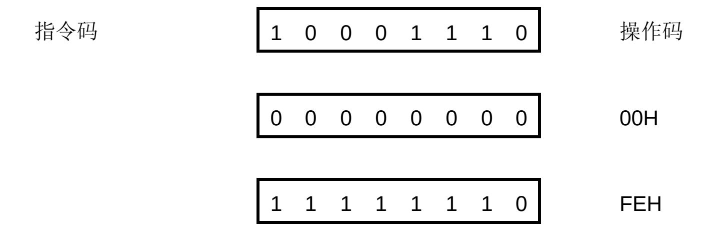
会根据指令字中的立即数作为操作数地址，对存储器进行直接寻址操作。

[操作数格式]

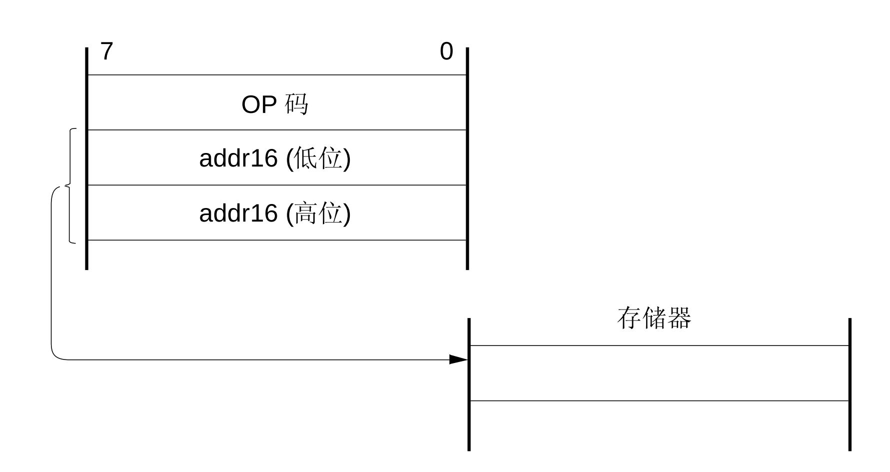
标识符	描述
addr16	标号或 16 位立即数

[描述举例]

MOV A，!0FE00H; 将!addr16 设置为 FE00H 时



[图示]



3.4.4 短直接寻址

[功能]

操作固定区域的存储器时，用指令中 8 位立即数进行直接寻址。

该方式的寻址范围是 FE20H 至 FF1FH 总共 256 字节的区域。内部高速 RAM 和特殊功能寄存器（SFR）分别映射到 FE20H 至 FEFFH 以及 FF00H 至 FF1FH 的区域。

可以采用短直接寻址方式的特殊功能寄存器（SFR）区域（FF00H 至 FF1FH）是整个特殊功能寄存器 SFR 区域的一部分。

程序中经常访问的端口、以及用作定时器和事件计数器的比较和捕捉寄存器都被映射到该区域。这些特殊功能寄存器（SFR）可以用较短字节和较少的时钟进行操作。

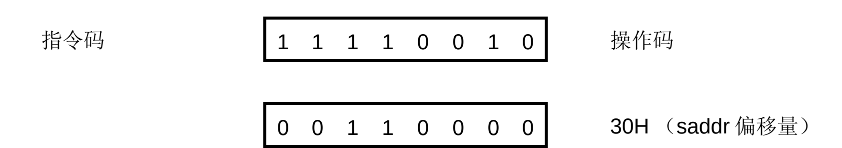
如果 8 位立即数是在 20H 和 FFH 之间，则有效地址的第 8 位被设置为 0；如果 8 位立即数是在 00H 与 1FH 之间，则有效地址的第 8 位被设置为 1。参见下面的 **[图示]** 。

[操作数格式]

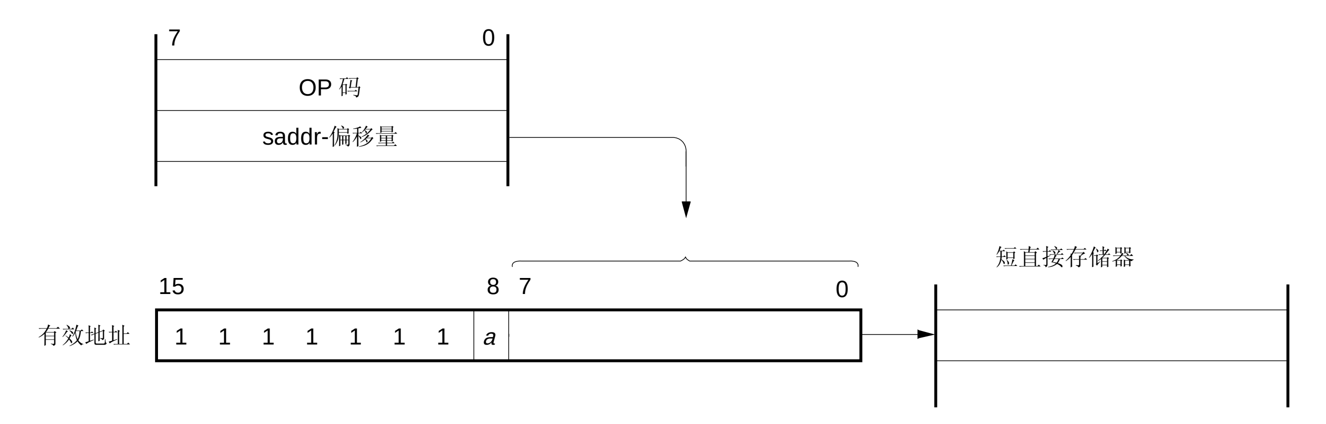
标识符	描述
saddr	表示标号或 FE20H 至 FF1FH 的立即数
saddrp	表示标号或 FE20H 至 FF1FH 的立即数 （仅使用偶地址）

[描述举例]

MOV 0FE30H，A; 将寄存器 A 的值赋给 saddr（FE30H）



[图示]



当 8 位立即数在 20H至FFH 之间时，α等于0 。

当 8 位立即数在 00H 至 1FH 之间时，α 等于 1。

3.4.5 特殊功能寄存器 （SFR） 寻址

[功能]

通过指令中的 8 位立即数对存储器映射的特殊功能寄存器 （SFR） 区域进行寻址。

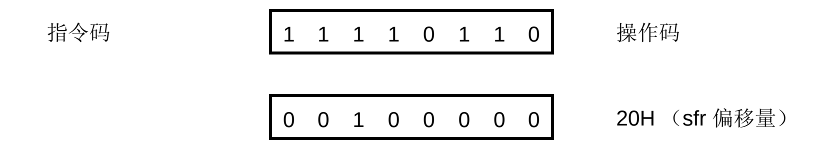
寻址区间为 FF00H 至 FFCFH 以及 FFE0H 至 FFFFH，共 240 字节。但是，映射在 FF00H 至 FF1FH 区间的特殊功能寄存器 （SFR） 则采用短直接寻址方式来访问。

[操作数格式]

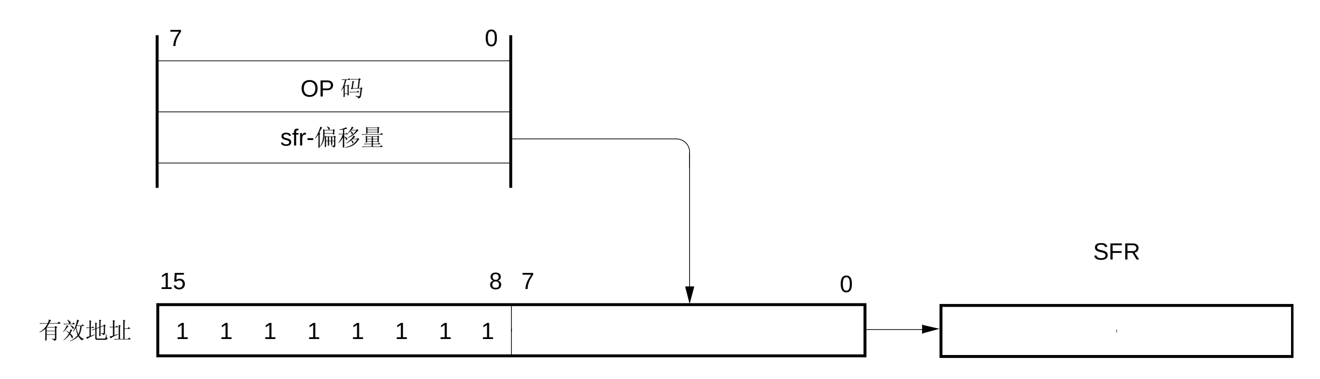
标识符	描述
sfr	特殊功能寄存器名称
sfrp	16 位可操作特殊功能寄存器名称（仅使用偶地址）

[描述举例]

MOV PM0，A; 选择 PM0（FF20H）作为 sfr



[图示]



3.4.6 寄存器间接寻址

[功能]

寄存器对的内容作为操作数地址，用来对存储器进行寻址，该寄存器对的内容由指令字中的寄存器对指定码和寄存器组选择标志（RBS0 和 RBS1）来指定。在整个存储器空间都可以执行这种寻址方式。

[操作数格式]

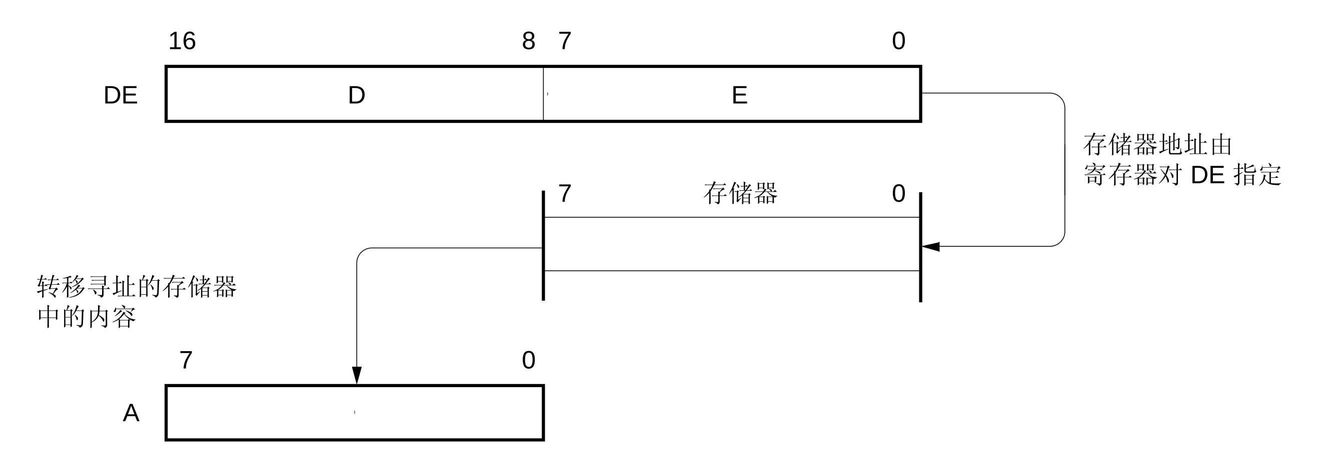
标识符	描述
–	[DE]，[HL]

[描述举例]

MOV A，[DE]; 选择 DE 作为寄存器对

操作码 1 0 0 0 0 1 0 1

[图示]



3.4.7 基址寻址

[功能]

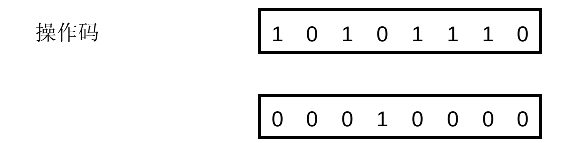
将 8 位立即数作为偏移量数据加上 HL 寄存器对的内容，根据相加结果寻址。由寄存器组选择标志（RBS0 和 RBS1）指定 HL 寄存器对作为基地址寄存器。将偏移量扩展为 16 位正数，来执行加法运算操作，第 16 位的进位被忽略。在整个存储器空间都可以执行这种寻址方式。

[操作数格式]

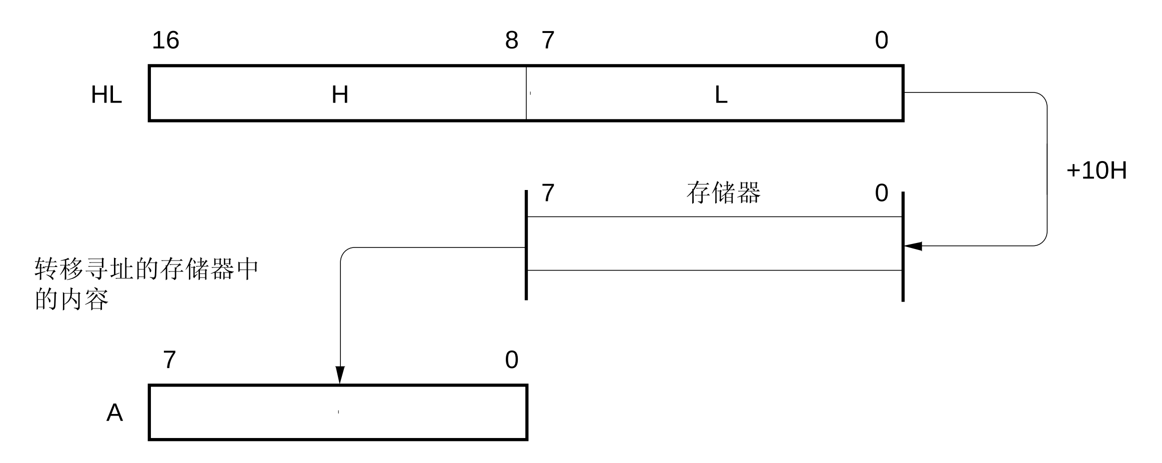
标识符	描述
–	[HL + 字节]

[描述举例]

MOV A，[HL + 10H]；设置字节为 10H 时



[图示]



3.4.8 基址变址寻址

[功能]

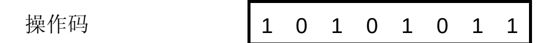
将指令字中指定的 B 或 C 寄存器的内容加上 HL 寄存器的内容，根据相加结果对存储器进行寻址。由寄存器组选择标志（RBS0 和 RBS1）指定 HL 寄存器对作为基地址寄存器。将 B 或 C 寄存器扩展为一个 16 位的正数来执行加法运算操作，第 16 位的进位被忽略。在整个存储器空间都可以执行这种寻址方式。

[操作数格式]

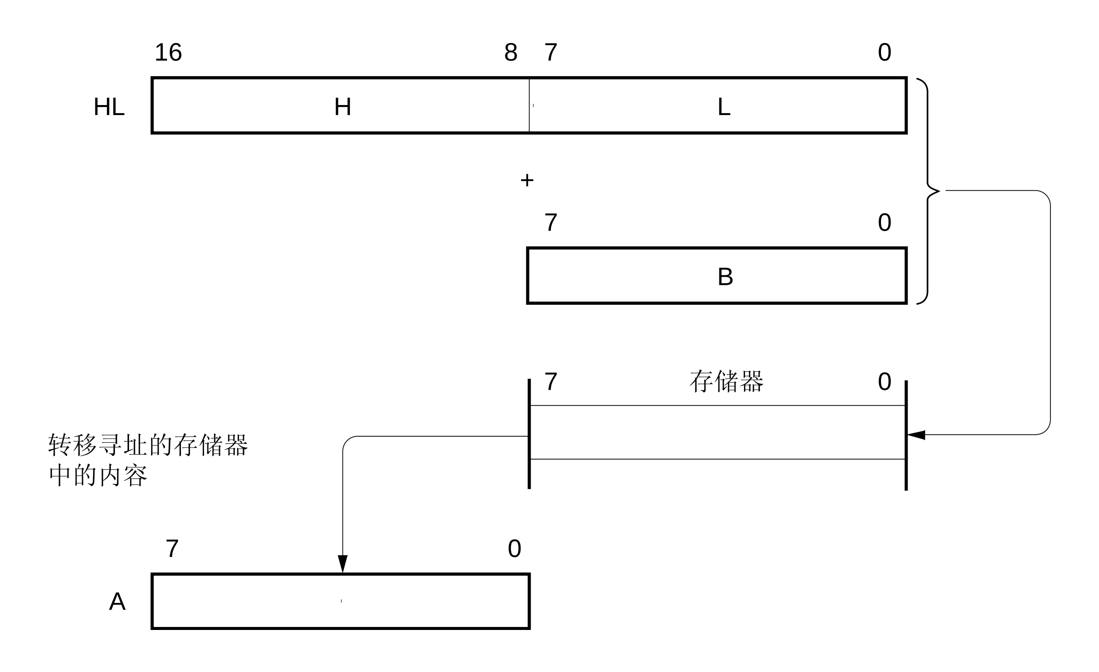
标识符	描述
–	[HL + B]，[HL + C]

[描述举例]

MOV A，[HL +B]; 选择 B 寄存器时



[图示]



3.4.9 堆栈寻址

[功能]

根据堆栈指针（SP）的内容对堆栈区域进行间接寻址。

当执行 PUSH、POP、子程序调用和返回指令时，或者产生中断请求时保存/恢复寄存器操作时，将自动采用这种寻址方式。

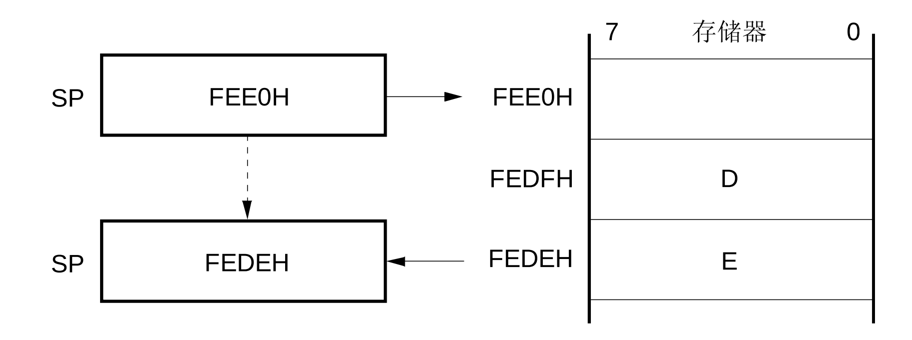
堆栈寻址方式仅适用于访问内部高速 RAM 区域。

[描述举例]

PUSH DE; 保存 DE 寄存器时

操作码 | 1 0 1 1 0 1 0 1

[图示]



第四章 端口功能

4.1 端口功能

有两种类型的引脚输入/输出缓冲器电源：AV_{REF} [注]和 V_{DD}。这些电源和引脚之间的关系显示如下。

表 4-1. 引脚 输入/输出 缓冲器电源

供电电源	相关引脚
AV_{REF} [注]	P20 至 P27
V_{DD}	除 P20 至 P27 的其它引脚

注　仅限μPD78F045x 和 78F046x。　PD78F044x 的供电电源是 V_{DD}。

78K0/LE3 产品提供的端口如图 4-1 所示，这些端口可以支持多种控制操作。每个端口的功能在表 4-2 中列出。

除了作为数字输入/输出端口外，这些端口还有复用功能。如需了解这些端口复用功能的详细信息，参见**第二章 引脚功能**。

图 4-1. 端口类型

表 4-2. 端口功能(1/2)

引脚名称	输入/输出	功能	复位后	复用功能
P11	输入/输出	端口 1 4 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。	输入端口	$\overline{SCK10}$
P12				SI10/RxD0/<RxD6>
P13				SO10/TxD0/<TxD6>
P14				INTP4
P20	输入/输出	端口 2 8 位 输入/输出 端口。 可以按位选择输入/输出模式。	数字输入端口	SEG31[注 1]/ANI0[注 2]/DS0–[注 3]
P21				SEG30[注 1]/ANI1[注 2]/DS0+[注 3]
P22				SEG29[注 1]/ANI2[注 2]/DS1–[注 3]
P23				SEG28[注 1]/ANI3[注 2]/DS1+[注 3]
P24				SEG27[注 1]/ANI4[注 2]/DS2–[注 3]
P25				SEG26[注 1]/ANI5[注 2]/DS2+[注 3]
P26				SEG25[注 1]/ANI6[注 2]/REF–[注 3]
P27				SEG24[注 1]/ANI7[注 2]/REF+[注 3]
P31	输入/输出	端口 3 4 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。	输入端口	TOH1/INTP3
P32				TOH0/MCGO
P33				TI000/RTCDIV/ RTCCL/BUZ/INTP2
P34				TI52/TI010/TO00/ RTC1HZ/INTP1
P40	输入/输出	端口 4 5 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。	输入端口	V_{LC3}/KR0
P41				RIN/KR1
P42				KR2
P43				TO51/TI51/KR3
P44				TO50/TI50/KR4
P80 至 P83	输入/输出	端口 8 4 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。	输入端口	SEG4 至 SEG7

注 **1.** 仅限μPD78F044x 和 78F045x 。
2. 仅限μPD78F045x 和 78F046x。
3. 仅限μPD78F046x。

备注 尖括号 (< >) 内的功能由输入转换控制寄存器 (ISC)分配。

表 **4-2. 端口功能(2/2)**

引脚名称	输入/输出	功能	复位后	复用功能
P100 至 P103	输入/输出	端口 10 4 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。	输入端口	SEG8 至 SEG11
P110，P111	输入/输出	端口 11 4 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。	输入端口	SEG12, SEG13
P112				SEG14/TxD6
P113				SEG15/RxD6
P120	输入/输出	端口 12. 1 位输入/输出端口和 4 位 输入端口。 可以按位选择输入/输出模式。 仅有 P120 可以通过软件设置指定内置上拉电阻的使用。	输入端口	INTP0/EXLVI
P121	输入			X1/OCD0A
P122				X2/EXCLK/OCD0B
P123				XT1
P124				XT2
P140 至 P143	输入/输出	端口 14 4 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。.	输入端口	SEG16 (KS0) 至 SEG19 (KS3)
P150 至 P153	输入/输出	端口 15 4 位 输入/输出 端口。 可以按位选择输入/输出模式。 通过软件设置，可以指定内置上拉电阻的使用。	输入端口	SEG20 (KS4) 至 SEG23 (KS7)

4.2 端口配置

端口包括如下硬件。

表 4-3. 端口配置

项目	配置
控制寄存器	端口模式寄存器(PM1 至 PM4，PM8，PM10 至 PM12，PM14，PM15) 端口寄存器(P1 至 P4，P8 ，P10 至 P12，P14，P15) 上拉电阻选择寄存器(PU1，PU3，PU4，PU8，PU10 至 PU12，PU14，PU15) 端口功能寄存器 1 (PF1) 端口功能寄存器 2 (PF2)[注 1] 端口功能寄存器 ALL (PFALL) A/D 端口配置寄存器 0 (ADPC0)[注 2]
端口	总计： 46
上拉电阻	总计： 34

注 **1.** 仅限μPD78F044x 和 78F045x。
2. 仅限μPD78F045x 和 78F046x 。

4.2.1 端口 1

端口 1 是具有输出锁存功能的 4 位输入/输出端口。使用端口模式寄存器 1(PM1)，可以按位选择端口 1 作为输入或输出模式。如果 P11 至 P14 作为输入端口，则可以通过上拉电阻选择寄存器 1(PU1)按位指定内部上拉电阻的使用。

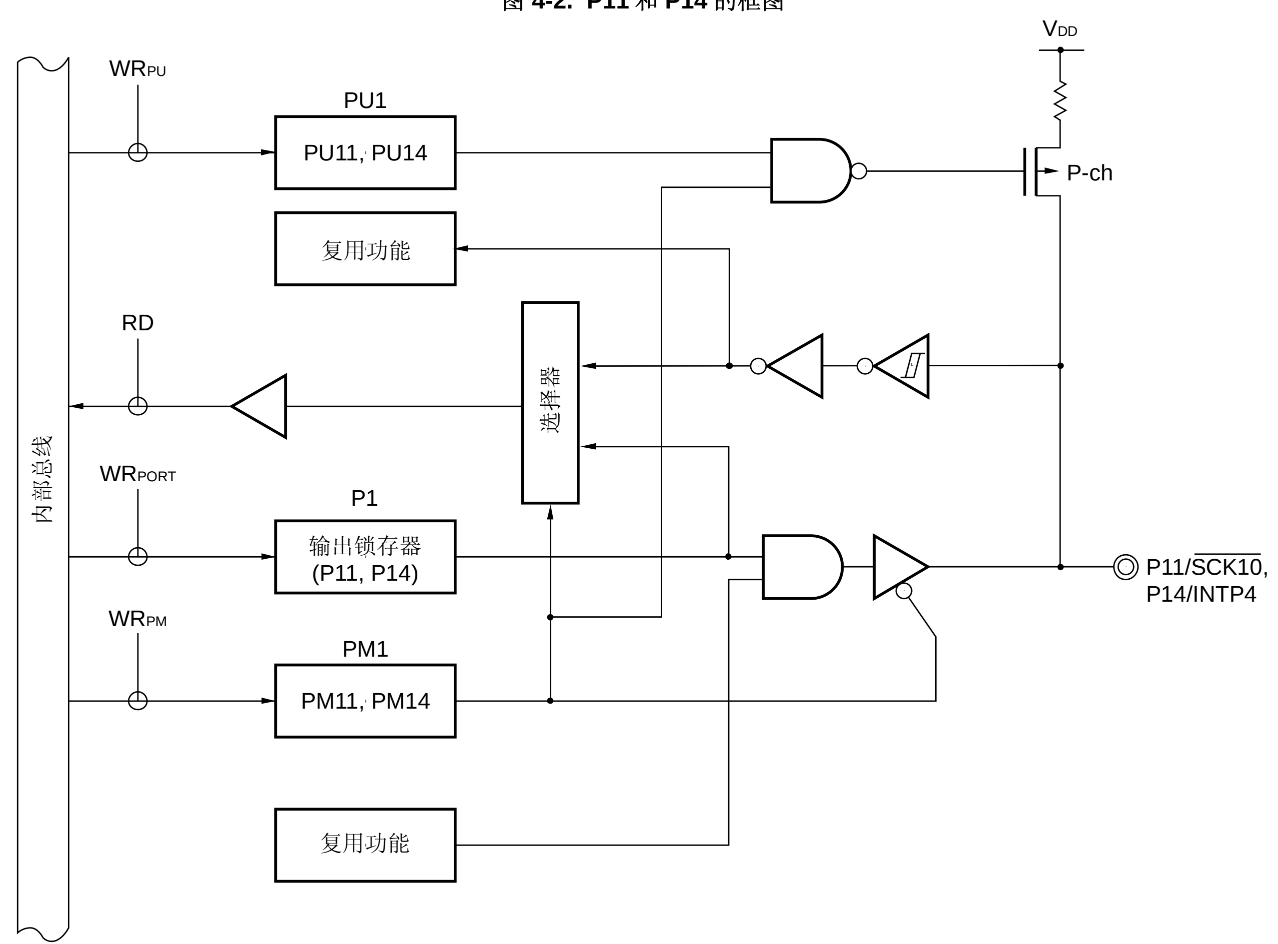
该端口也可以用作串行时钟输入/输出、串行接口数据输入/输出和可屏蔽外部中断输入。

复位信号的产生会将端口 1 设置为输入模式。

图 4-2 至 4-4 显示端口 1 的框图。

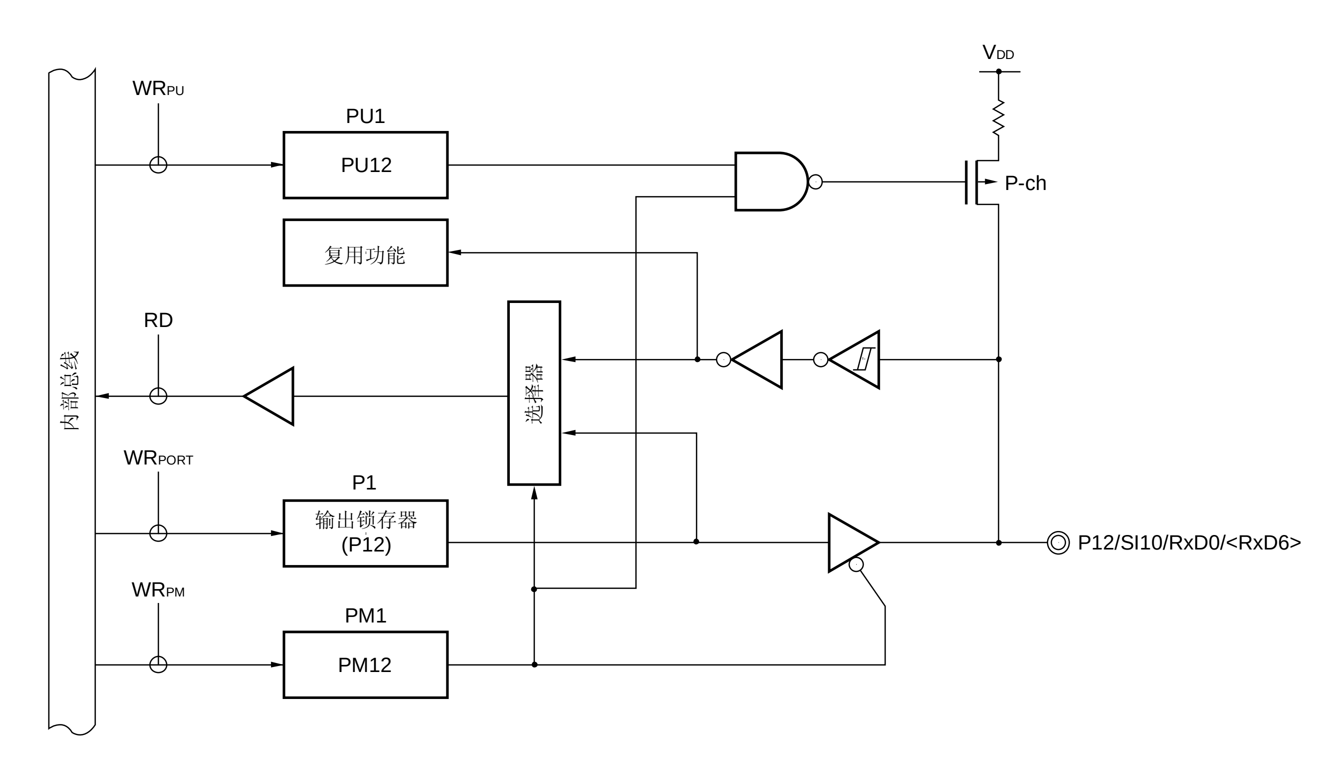
注意事项 如果 **P11/$\overline{\text{SCK10}}$，P12/SI10/RxD0/<RxD6>**和 **P13/SO10/TxD0/<TxD6>**要作为通用端口使用，则将串行操作模式寄存器 **10(CSIM10)**和串行时钟选择寄存器 **10(CSIC10)**设置为默认状态（**00H**）。

图 4-2. P11 和 P14 的框图



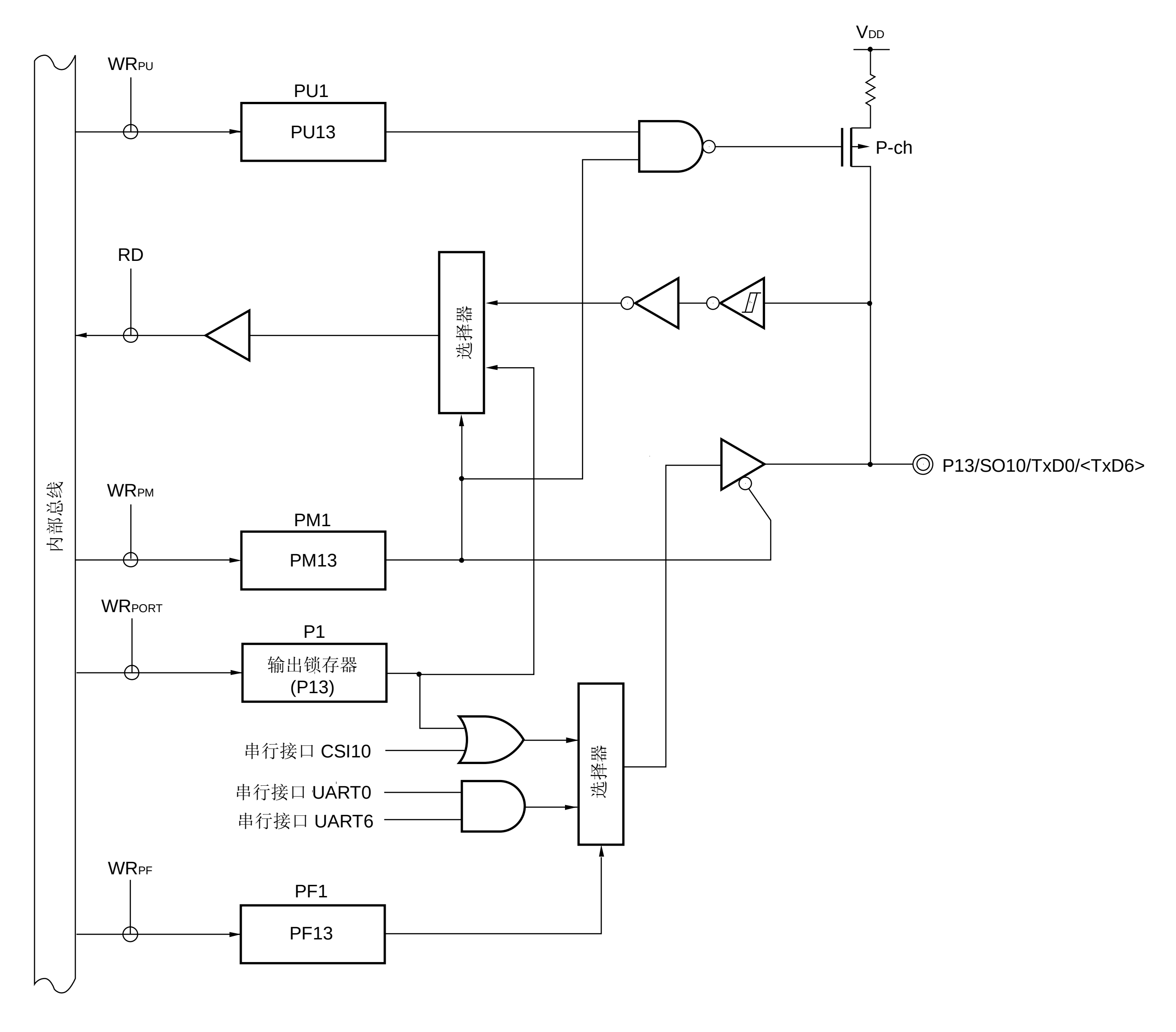
P1: 端口寄存器 1
PU1: 上拉电阻选择寄存器 1
PM1: 端口模式寄存器 1
RD: 读信号
WR××: 写信号

图 **4-3. P12** 的框图



P1：　端口寄存器 1
PU1：　上拉电阻选择寄存器 1
PM1：　端口模式寄存器 1
RD：　读信号
WR××：　写信号

图 4-4. P13 的框图



P1： 端口寄存器 1
PU1： 上拉电阻选择寄存器 1
PM1： 端口模式寄存器 1
RD： 读信号
WR××： 写信号

4.2.2 端口 2

端口 2 是具有输出锁存功能的 8 位输入/输出端口。使用端口模式寄存器 2(PM2)，可以按位选择端口 2 为输入或输出模式。

该端口也可以用作 10 位逐次逼近型 A/D 转换器、16 位 ΔΣ 型 A/D 转换器模拟输入和 segment 输出。

要将 P20/ANI0/DS0–，P21/ANI1/DS0+，P22/ANI2/DS1–，P23/ANI3/DS1+，P24/ANI4/DS2–，P25/ANI5/DS2+，P26/ANI6/REF–和 P27/ANI7/REF+ 作为数字输入引脚，由端口功能寄存器 2 (PF2)设置端口功能 (除 segment 输出外)，使用 ADPC0 设置数字输入/输出，且使用 PM2 设置为输入模式。从低位开始使用这些引脚。

要将 P20/ANI0/DS0–，P21/ANI1/DS0+，P22/ANI2/DS1–，P23/ANI3/DS1+，P24/ANI4/DS2–，P25/ANI5/DS2+，P26/ANI6/REF–和 P27/ANI7/REF+作为数字输出引脚，由端口功能寄存器 2 (PF2)设置端口功能 (除 segment 输出外)，使用 ADPC0 设置数字输入/输出，且使用 PM2 设置为输出模式。从低位开始使用这些引脚。

复位信号的产生会将端口 2 设置为输入模式。

图 4-5 显示端口 2 的框图。

表 4-4. P20/SEG31[注1]/ANI0[注2]/DS0–[注3] 至 P27/SEG24[注1]/ANI7[注2]/REF+[注3] 引脚的功能设置

PF2	ADPC0	PM2	ADS	ADDCTL0	P20/SEG31[注1]/ANI0[注2]/DS0–N[注3] 至 P27/SEG24[注1]/ANI7[注2]/REF+[注3] 引脚
数字/模拟选择	模拟输入选择	输入模式	不选择 ANI	不选择 DSn±.	模拟输入 (不被转换)
			选择 ANI	不选择 DSn±.	模拟输入 (由逐次逼近型 A/D 转换器进行转换)
			不选择 ANI	选择 DSn±.	模拟输入 (由ΔΣ型 A/D 转换器进行转换)
			选择 ANI	选择 DSn±.	禁止设置
		输出模式	–		禁止设置
	数字输入/输出选择	输入模式	–		数字输入
		输出模式	–		数字输出
SEG 输出选择[注1]	–	–	–		Segment 输出选择[注1]

注 **1.** 仅限μPD78F044X 和 78F045x 。

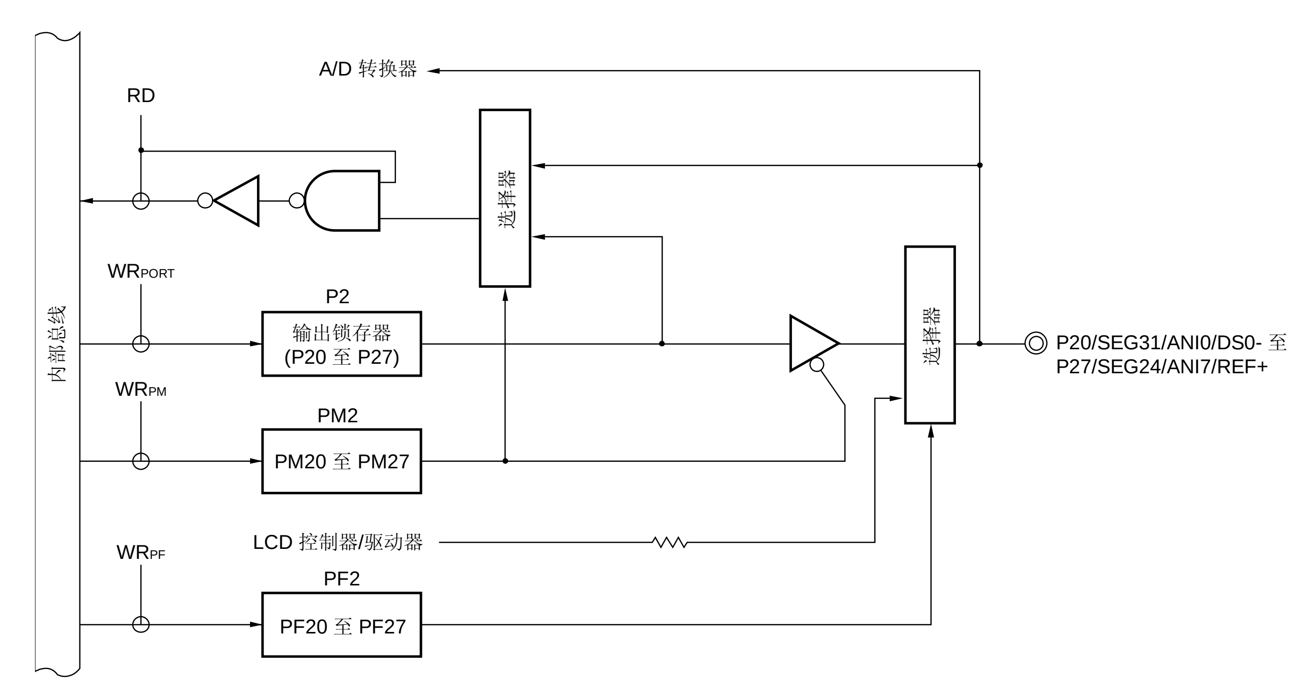
2. 仅限μPD78F045x 和 78F046x 。

3. 仅限μPD78F046x 。

备注 n = 0 至 2

<R>

图 **4-5. P20** 至 **P27** 的框图



P2：　端口寄存器 2
PM2：　端口模式寄存器 2
PF2：　端口功能寄存器 2
RD：　读信号
WR××：　写信号

4.2.3 端口 3

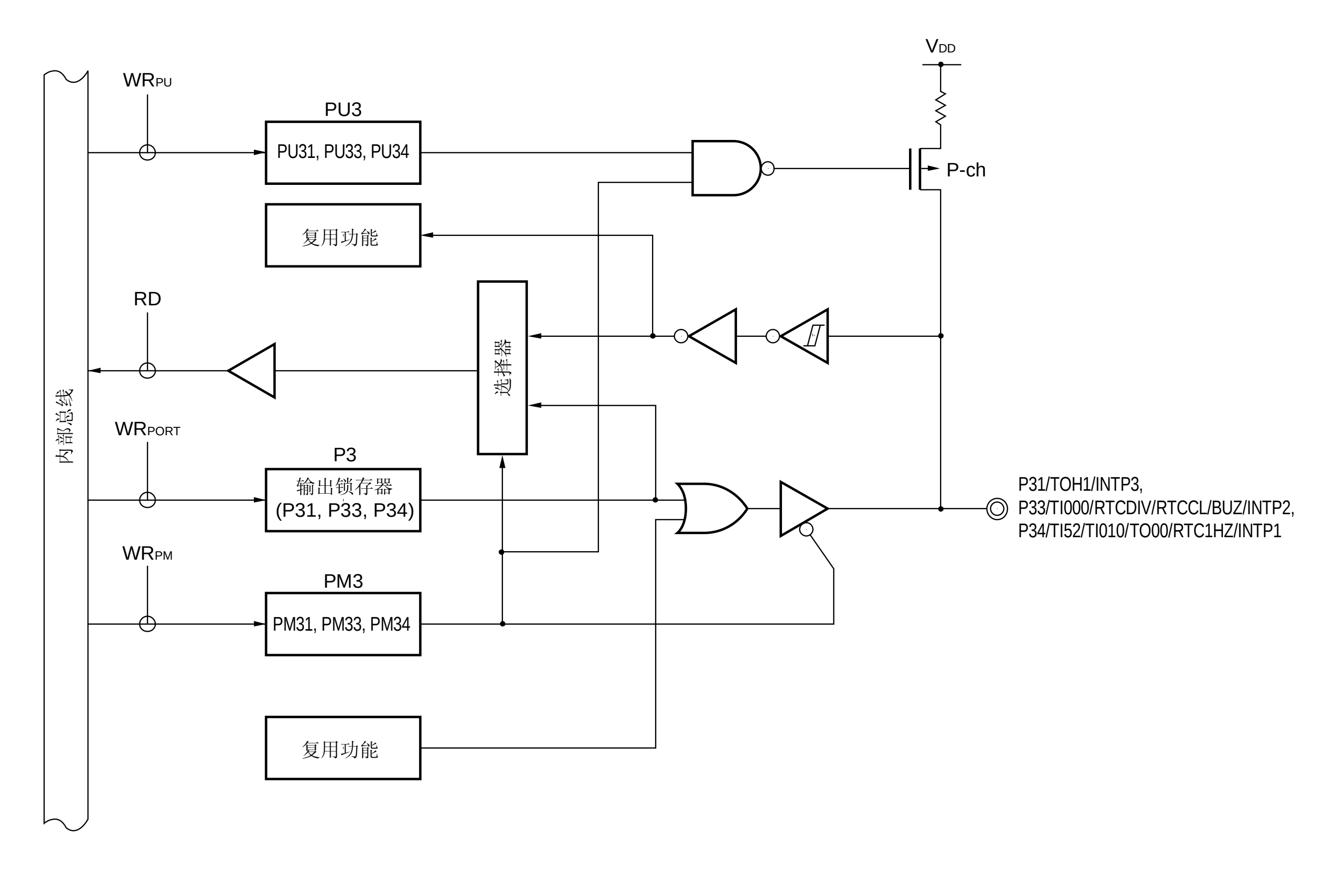
端口 3 是具有输出锁存功能的 4 位输入/输出端口。使用端口模式寄存器 3(PM3)，可以按位选择端口 3 为输入或输出模式。如果 P31 至 P34 作为输入端口，则可以通过上拉电阻选择寄存器 3(PU3)按位指定内部上拉电阻的使用。

该端口也可以用作外部中断请求输入、定时器输入/输出、曼彻斯特编码发生器输出、实时计数器输出和蜂鸣器输出。

复位信号的产生会将端口 3 设置为输入模式。

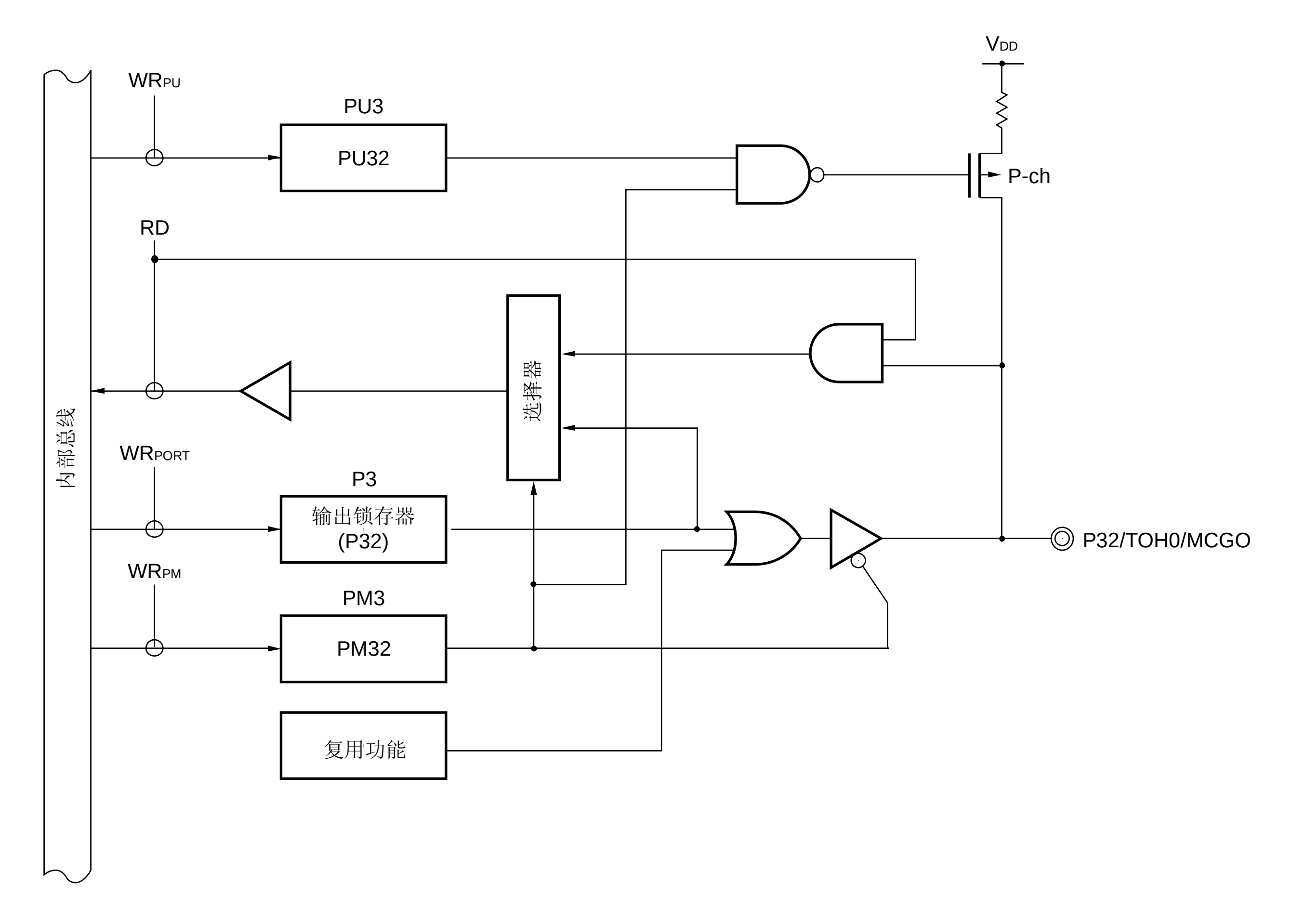
图 4-6 至 4-7 显示端口 3 的框图。

图 4-6. P31，P33，P34 的框图



P3： 端口寄存器 3
PU3： 上拉电阻选择寄存器 3
PM3： 端口模式寄存器 3
RD： 读信号
WR××： 写信号

图 4-7. P32 框图



P4： 端口寄存器 4
PU4： 上拉电阻选择寄存器 4
PM4： 端口模式寄存器 4
RD： 读信号
WR××： 写信号

4.2.4 端口 4

端口 4 是具有输出锁存功能的 5 位输入/输出端口。使用端口模式寄存器 4(PM4)，可以按位选择端口 4 为输入或输出模式。如果 P40 至 P44 作为输入端口，则可以通过上拉电阻选择寄存器 4(PU4)按位指定内部上拉电阻的使用。

该端口也可以用作按键中断输入、segment 键扫描输入、定时器输入/输出、遥控接收数据输入和 LCD 的驱动供电电源。

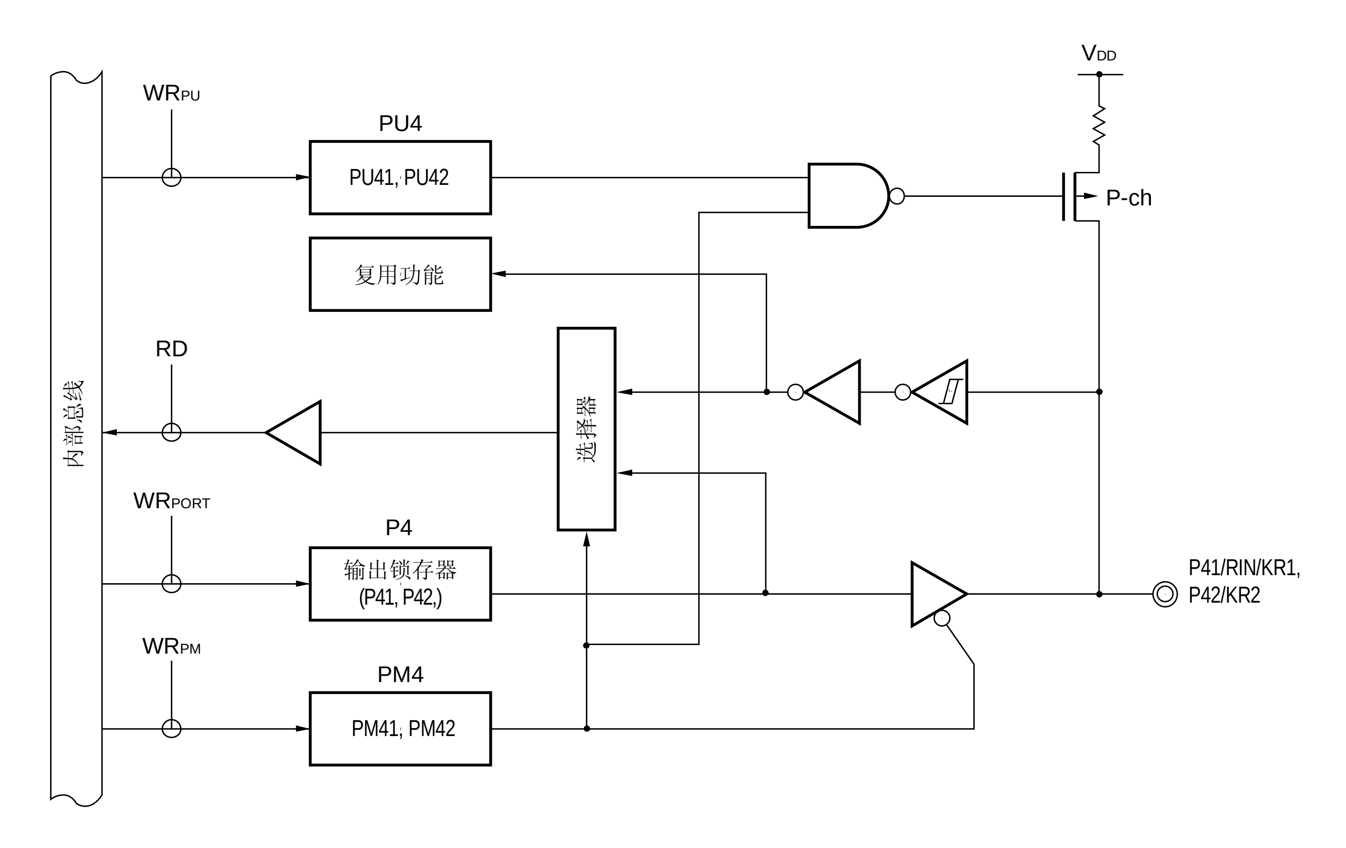
复位信号的产生会将端口 4 设置为输入模式。

图 4-8 至 4-10 显示端口 4 的框图。

图 4-8. P40 框图

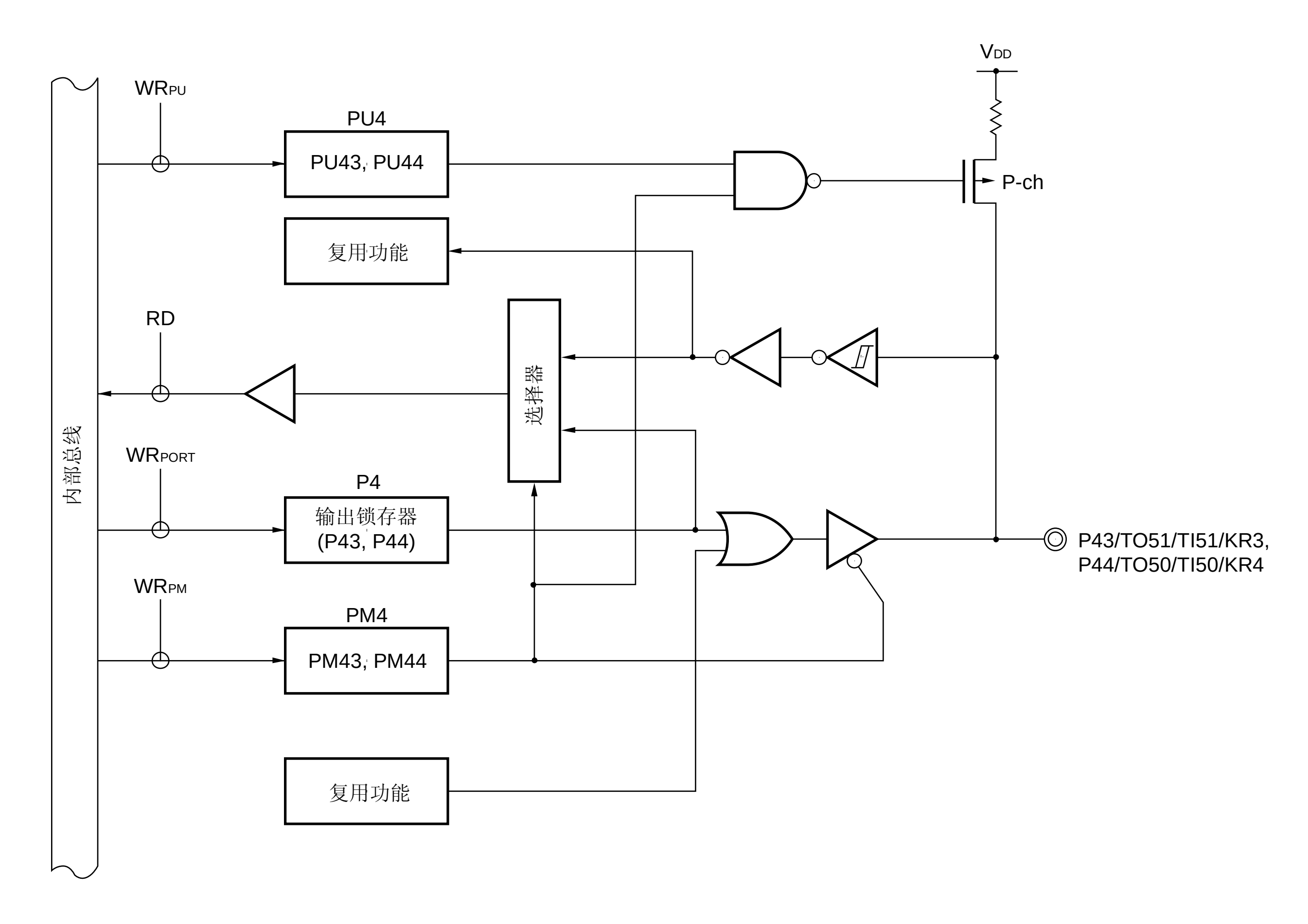
P4： 端口寄存器 4
PU4： 上拉电阻选择寄存器 4
PM4： 端口模式寄存器 4
LCDM：LCD 显示模式寄存器
RD： 读信号
WR××： 写信号

图 4-9. P41 和 P42 的框图



P4： 端口寄存器 4
PU4： 上拉电阻选择寄存器 4
PM4： 端口模式寄存器 4
RD： 读信号
WR××： 写信号

图 **4-10. P43** 和 **P44** 的框图



P4：　端口寄存器 4
PU4：　上拉电阻选择寄存器 4
PM4：　端口模式寄存器 4
RD：　读信号
WR××：　写信号

4.2.5 端口 8

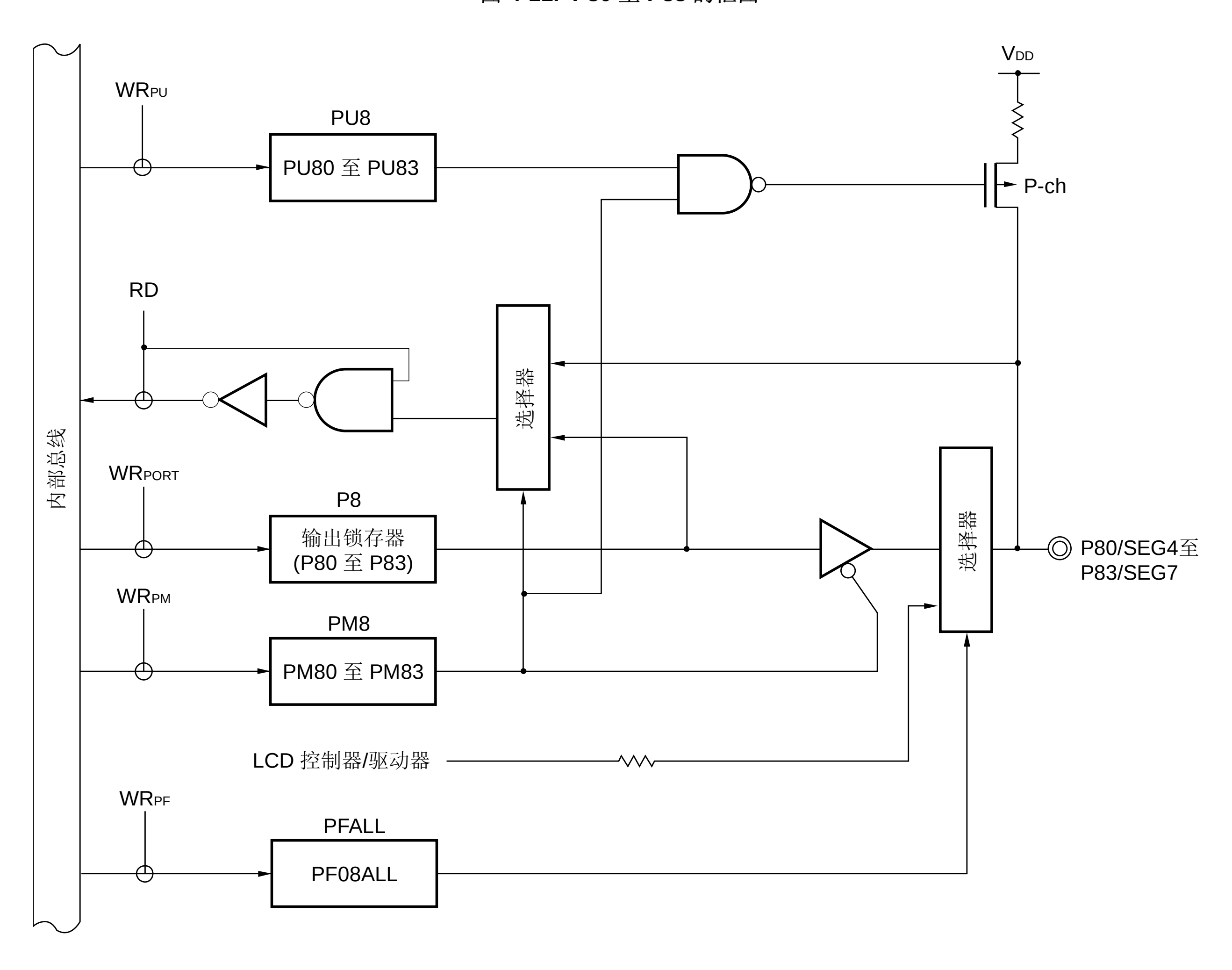
端口 8 是具有输出锁存功能的 4 位输入/输出端口。使用端口模式寄存器 8(PM8)，可以按位选择端口 8 为输入或输出模式。如果 P80 至 P83 作为输入端口，则可以通过上拉电阻选择寄存器 8(PU8)按位指定内部上拉电阻的使用。

该端口也可以用作 segment 输出。

复位信号的产生会将端口 8 设置为输入模式。

图 4-11 显示端口 8 的框图。

图 4-11. P80 至 P83 的框图



P8： 端口寄存器 8
PU8： 上拉电阻选择寄存器 8
PM8： 端口模式寄存器 8
PFALL： 端口功能寄存器 ALL
RD： 读信号
WR××： 写信号

4.2.6 端口 10

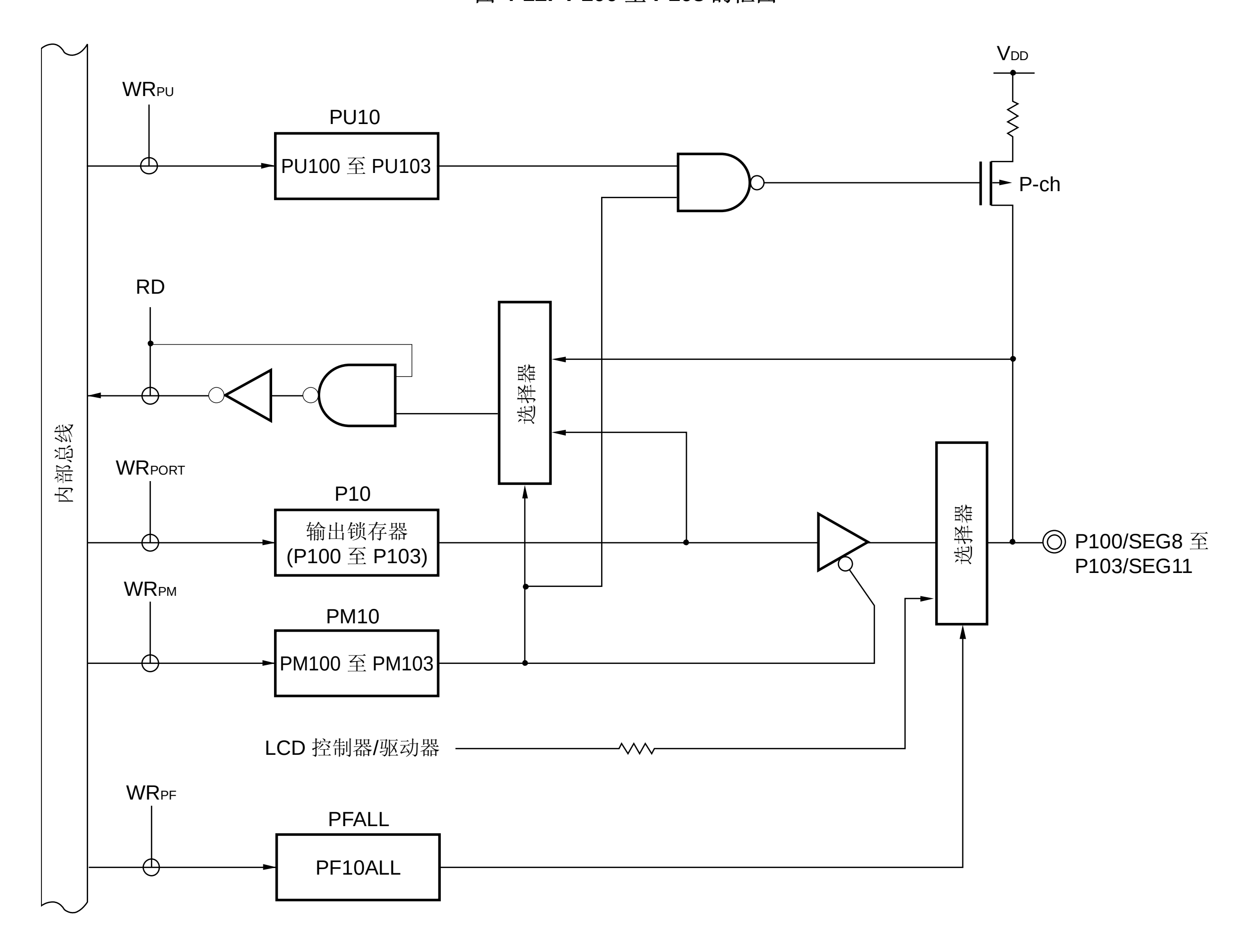
端口 10 是具有输出锁存功能的 4 位输入/输出端口。使用端口模式寄存器 10(PM10)，可以按位选择端口 10 为输入或输出模式。如果 P100 至 P103 作为输入端口，则可以通过上拉电阻选择寄存器 10(PU10)按位指定内部上拉电阻的使用。

该端口也可以用作 segment 输出。

复位信号的产生会将端口 10 设置为输入模式。

图 4-12 显示端口 10 的框图。

图 4-12. P100 至 P103 的框图



P10：　端口寄存器 10

PU10：　上拉电阻选择寄存器 10

PM10：　端口模式寄存器 10

PFALL：　端口功能寄存器 ALL

RD：　读信号

WR××：　写信号

4.2.7 端口 11

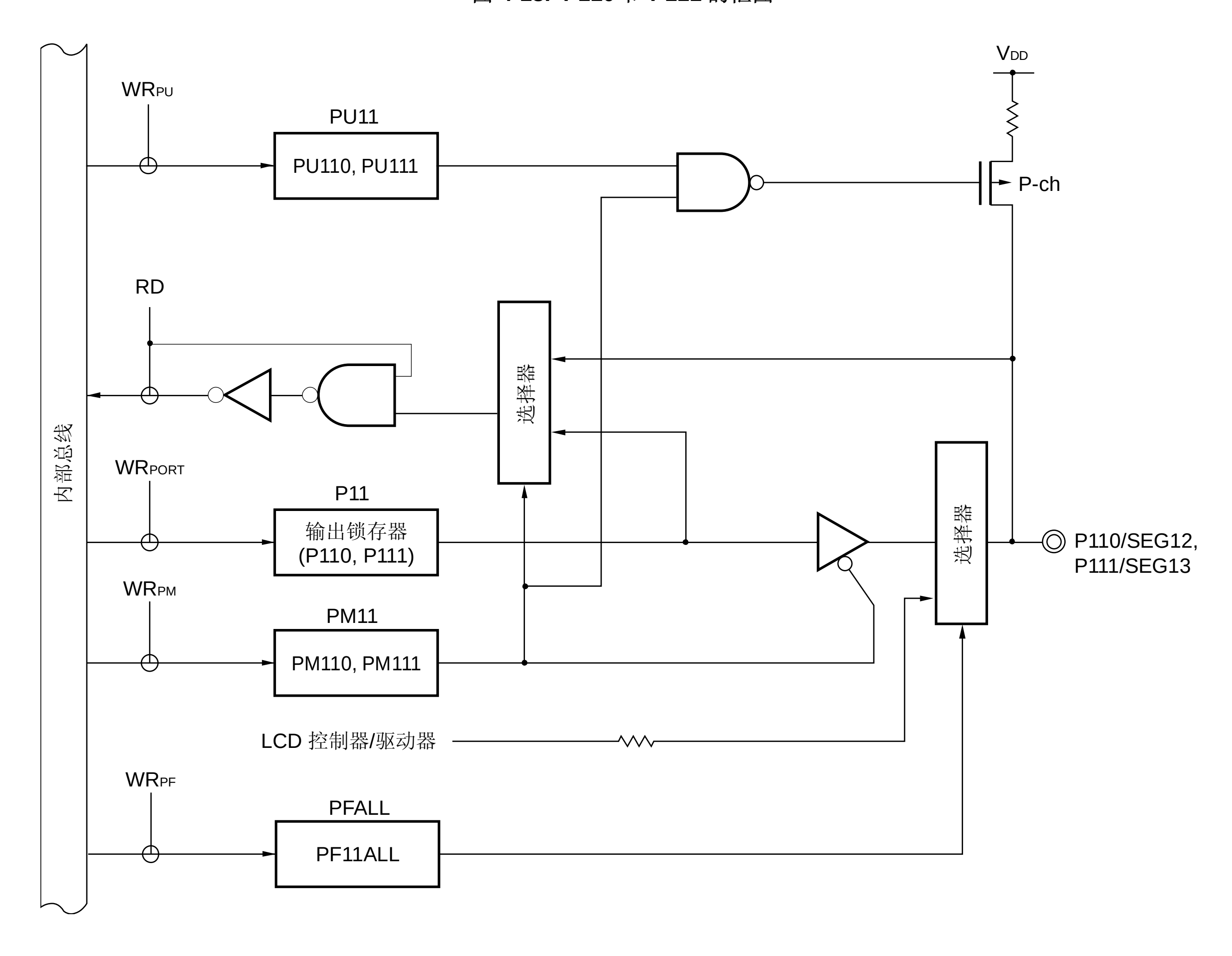
端口 11 是具有输出锁存功能的 4 位输入/输出端口。使用端口模式寄存器 11(PM11)，可以按位选择端口 11 为输入或输出模式。如果 P110 至 P113 作为输入端口，则可以通过上拉电阻选择寄存器 11(PU11)按位指定内部上拉电阻的使用。

该端口也可以用作 segment 输出和串行接口数据输入/输出。

复位信号的产生会将端口 11 设置为输入模式。

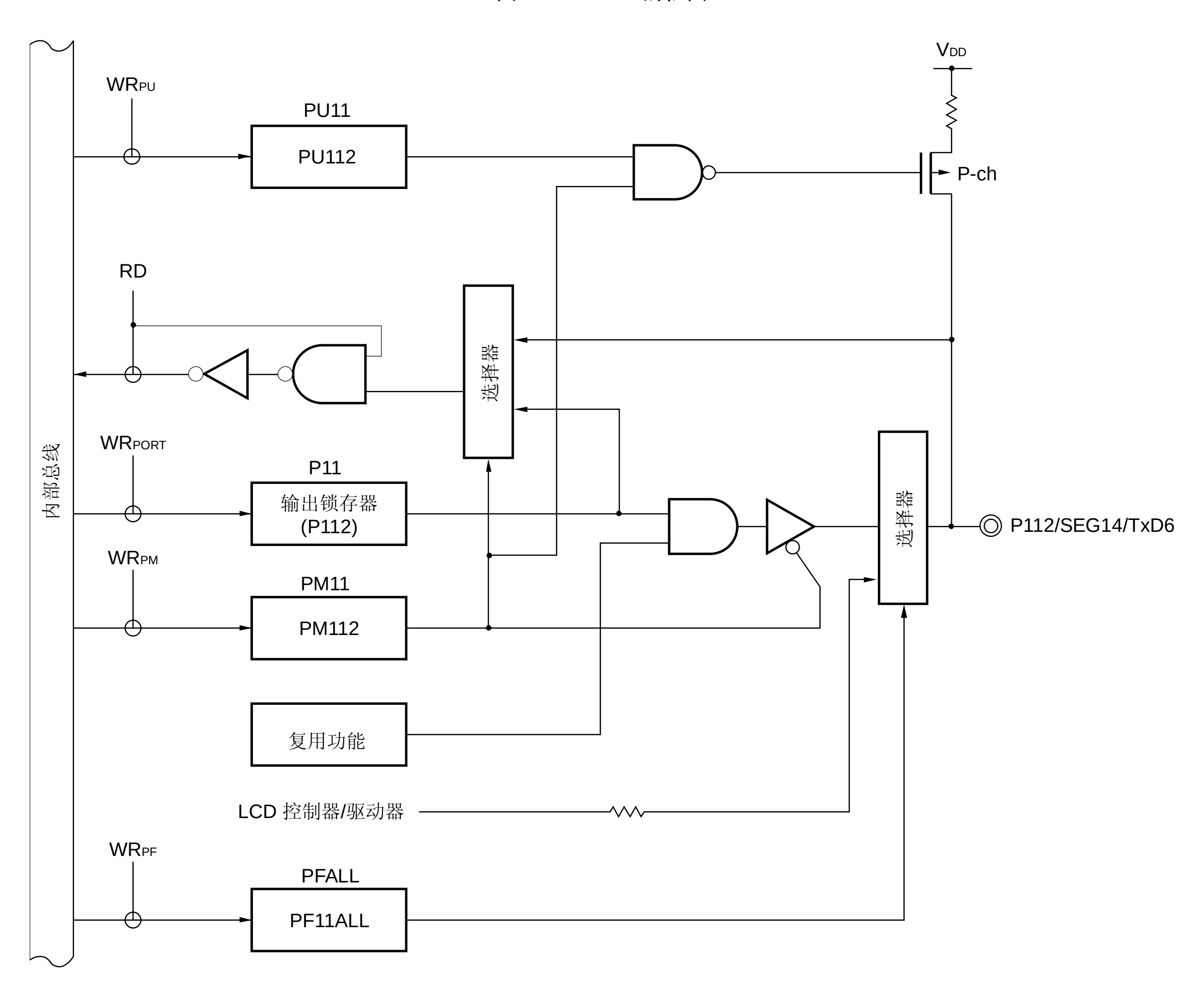
图 4-13 至 4-15 显示端口 11 的框图。

图 4-13. P110 和 P111 的框图



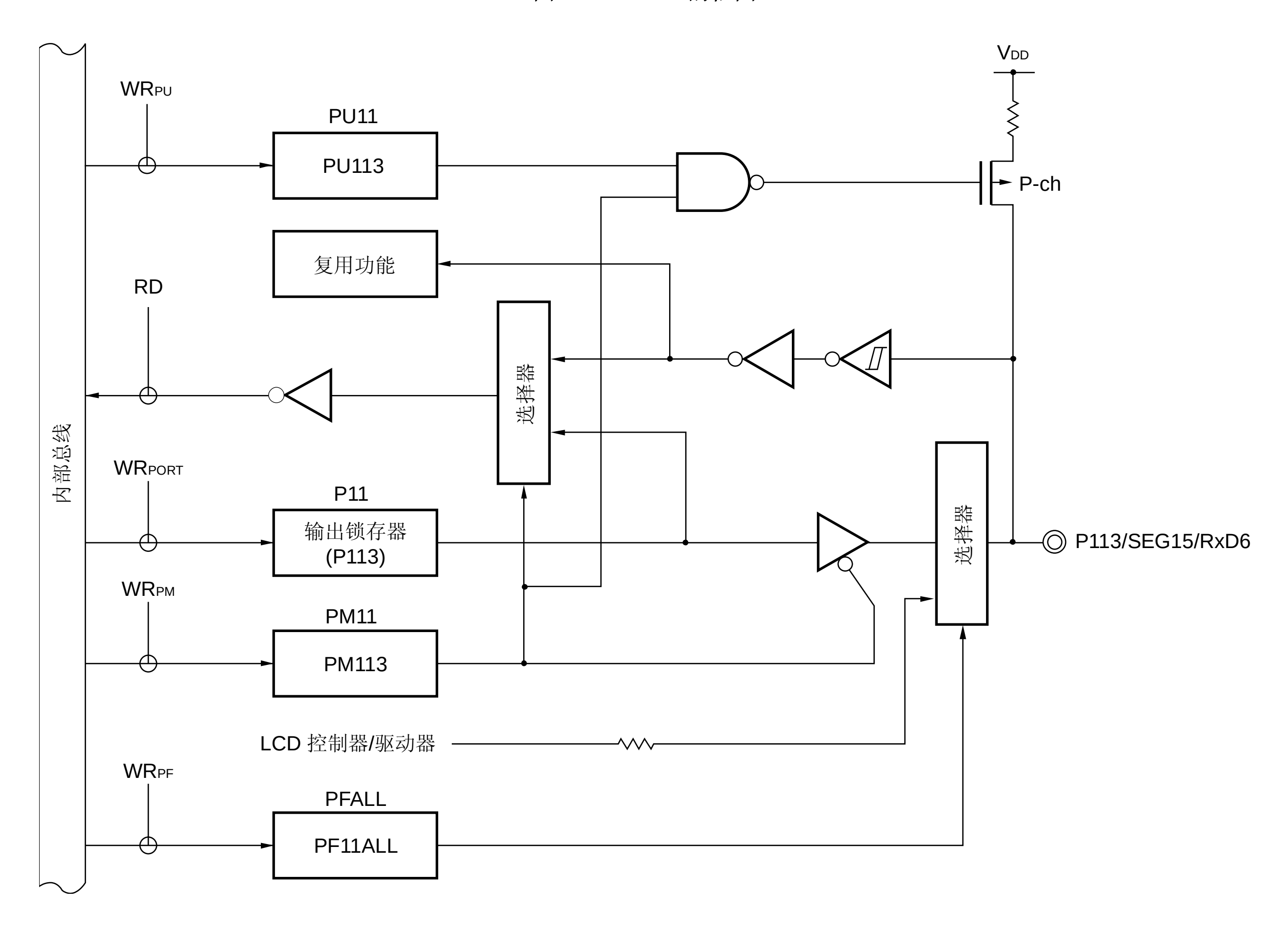
P11：　端口寄存器 11
PU11：　上拉电阻选择寄存器 11
PM11：　端口模式寄存器 11
PFALL：　端口功能寄存器 ALL
RD：　读信号
WR××：　写信号

图 **4-14. P112** 的框图



P11：　端口寄存器 11
PU11：　上拉电阻选择寄存器 11
PM11：　端口模式寄存器 11
PFALL：　端口功能寄存器 ALL
RD：　读信号
WR××：　写信号

图 4-15. P113 的框图



P11：　端口寄存器 11

PU11：　上拉电阻选择寄存器 11

PM11：　端口模式寄存器 11

PFALL：　端口功能寄存器 ALL

RD：　读信号

WR××：　写信号

4.2.8 端口 12

端口 12 是具有输出锁存功能的 1 位输入/输出端口和 4 位输入端口。仅 P120 可以使用端口模式寄存器 12(PM12)，按位选择端口为输入或输出模式。如果 P120 作为输入端口，则可以通过上拉电阻选择寄存器 12(PU12)指定内部上拉电阻的使用。

该端口也可以用作外部中断请求输入、外部低电压检测的电位输入、主系统时钟的振荡器连接、副系统时钟的振荡器连接和主系统时钟的外部时钟输入。

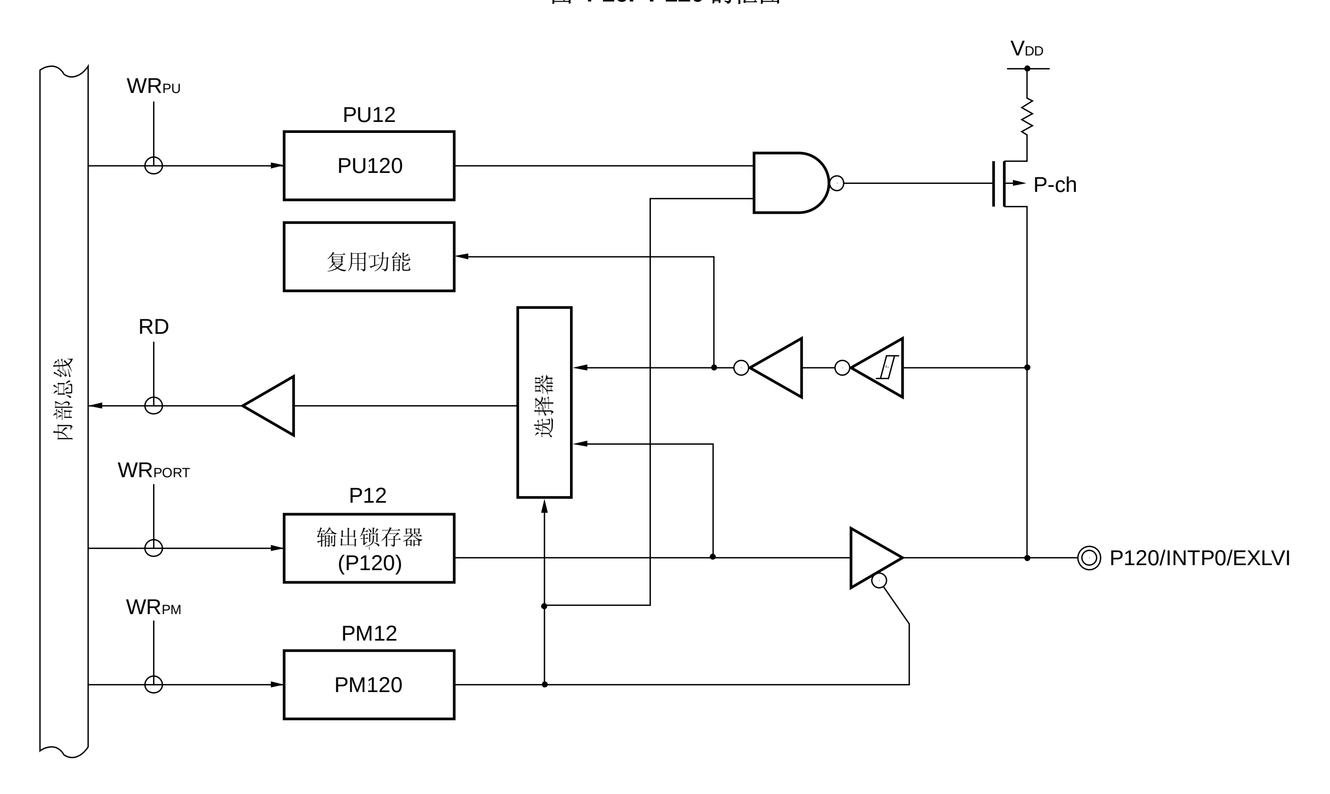
复位信号的产生会将端口 12 设置为输入模式。

图 4-16 至 4-18 显示端口 12 的框图。

注意事项 **当 P121 至 P124 引脚用于连接主系统时钟（X1，X2）或副系统时钟（XT1，XT2）的振荡器时，或者输入外部时钟（EXCLK）作为主系统时钟时，必须使用时钟操作模式选择寄存器（OSCCTL）对 X1 振荡模式、XT1 振荡模式或者外部时钟输入模式进行设置（如需了解详细信息，参见 5.3 (1) 时钟操作模式选择寄存器（OSCCTL）和（3）副系统时钟引脚操作模式设置）。OSCCTL 的复位值为 00H（P121 至 P124 都用作输入端口引脚）。**

备注 **P121 和 P122 能够用于片上调试模式设置引脚 (OCD0A，OCD0B) 。详情参见第二十八章　片上调试功能。**

图 **4-16. P120** 的框图



P12： 端口寄存器 12

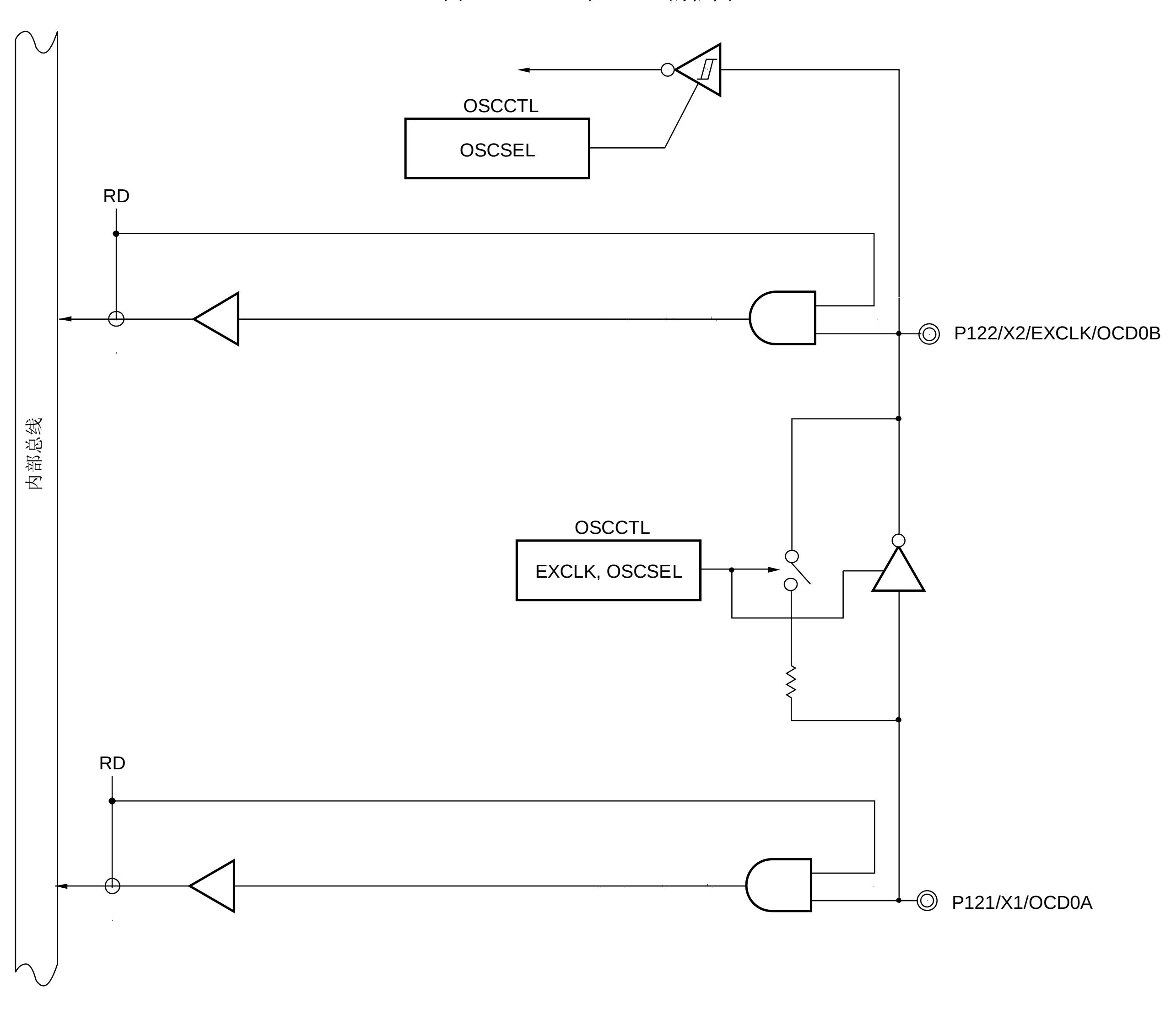
PU12： 上拉电阻选择寄存器 12

PM12： 端口模式寄存器 12

RD： 读信号

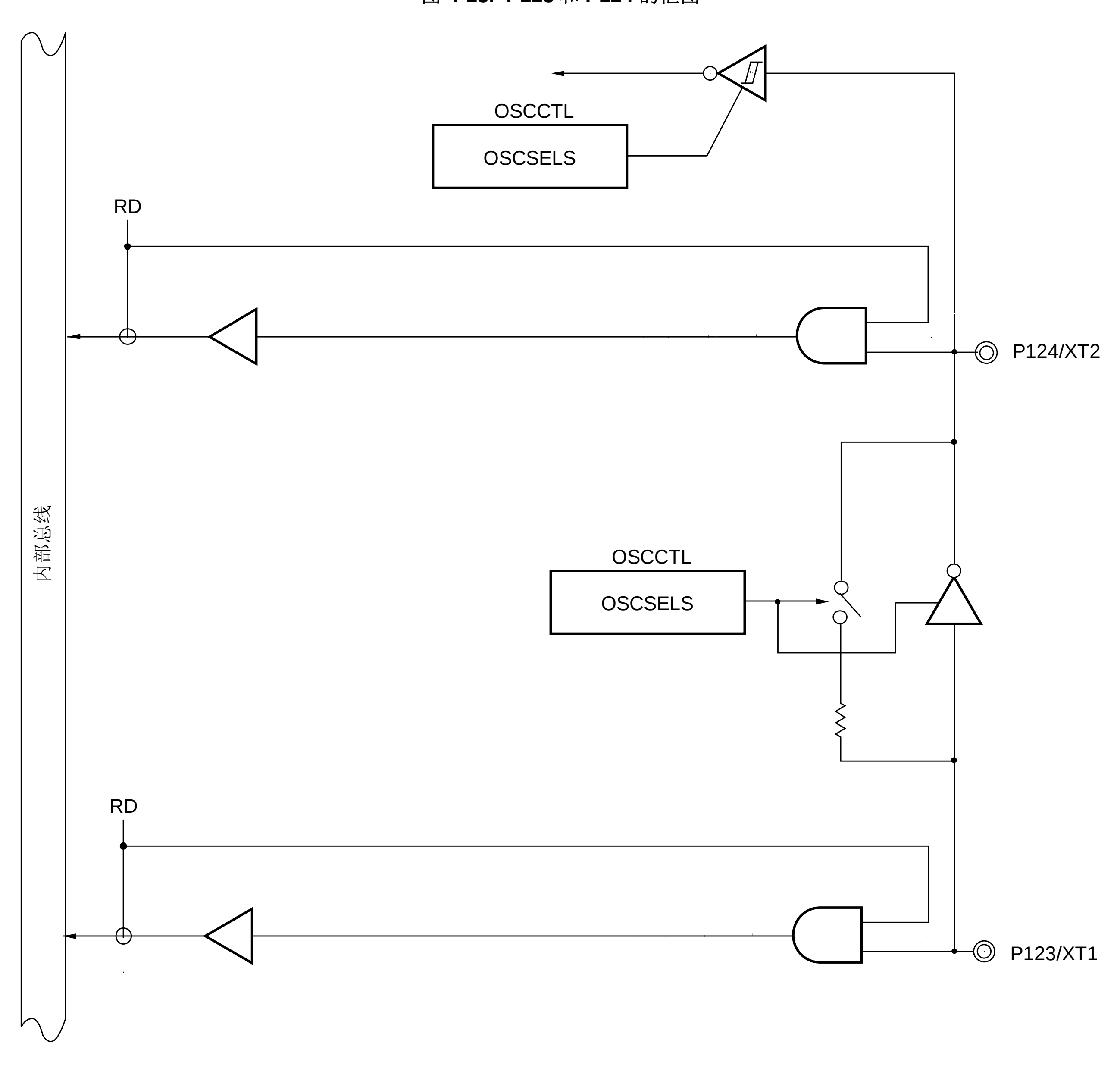
WR××： 写信号

图 **4-17. P121** 和 **P122** 的框图



OSCCTL： 时钟操作模式选择寄存器
RD： 读信号

图 **4-18. P123** 和 **P124** 的框图



OSCCTL： 时钟操作模式选择寄存器
RD： 读信号

4.2.9 端口 14

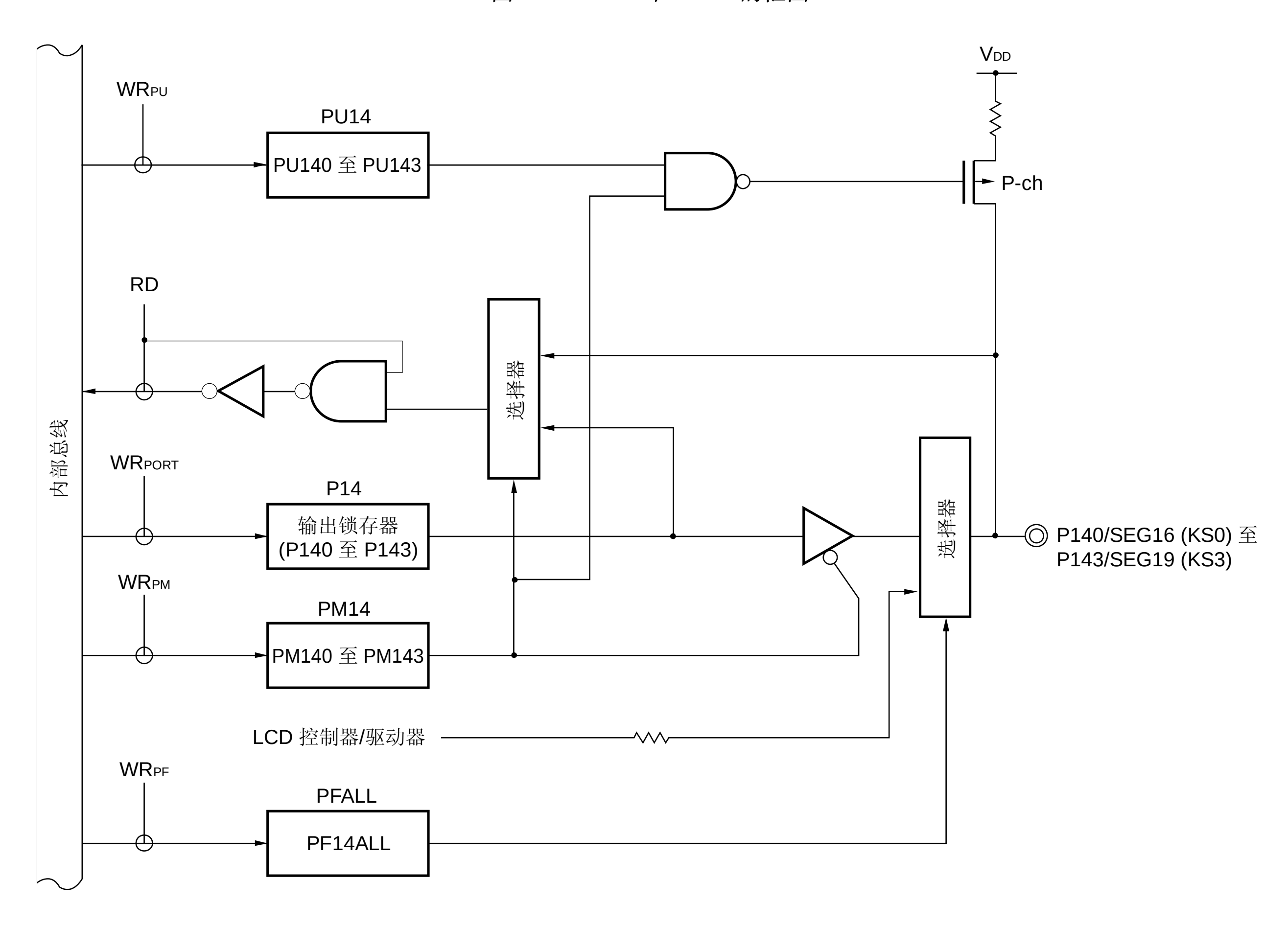
端口 14 是具有输出锁存功能的 4 位输入/输出端口。使用端口模式寄存器 14(PM14)，可以按位选择端口 14 为输入或输出模式。如果 P140 至 P143 作为输入端口，则可以通过上拉电阻选择寄存器 14(PU14)按位指定内部上拉电阻的使用。

该端口也可以用作 segment 输出。

复位信号的产生会将端口 14 设置为输入模式。

图 4-19 显示端口 14 的框图。

图 4-19. P140 和 P143 的框图



P14：　端口寄存器 14
PU14：　上拉电阻选择寄存器 14
PM14：　端口模式寄存器 14
PFALL：　端口功能寄存器 ALL
RD：　读信号
WR××：　写信号

4.2.10 端口 15

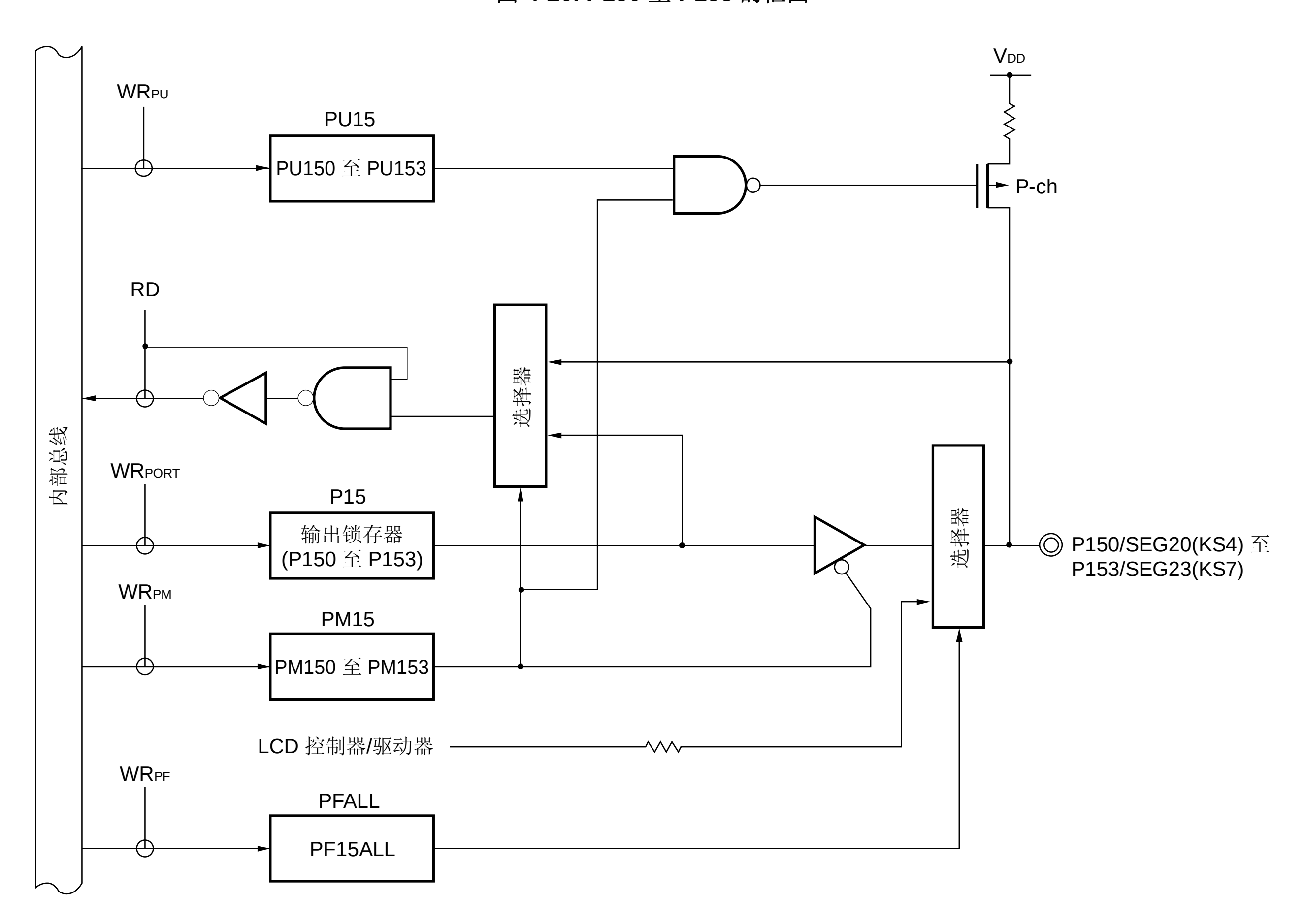
端口 15 是具有输出锁存功能的 4 位输入/输出端口。使用端口模式寄存器 15(PM15)，可以按位选择端口 15 为输入或输出模式。如果 P150 至 P153 作为输入端口，则可以通过上拉电阻选择寄存器 15(PU15)按位指定内部上拉电阻的使用。

该端口也可以用作 segment 输出。

复位信号的产生会将端口 15 设置为输入模式。

图 4-20 显示端口 15 的框图。

图 4-20. P150 至 P153 的框图



P15： 端口寄存器 15
PU15： 上拉电阻选择寄存器 15
PM15： 端口模式寄存器 15
PFALL： 端口功能寄存器 ALL
RD： 读信号
WR××： 写信号

4.3 控制端口功能的寄存器

端口功能由如下七种类型的寄存器控制。

- 端口模式寄存器(PM1 至 PM4，PM8，PM10 至 PM12，PM14，PM15)
- 端口寄存器(P1 至 P4，P8 ，P10 至 P12，P14，P15)
- 上拉电阻选择寄存器(PU1，PU3，PU4，PU8，PU10 至 PU12，PU14，PU15)
- 端口功能寄存器 1 (PF1)
- 端口功能寄存器 2 (PF2)[注 1]
- 端口功能寄存器 ALL (PFALL)
- A/D 端口配置寄存器 0 (ADPC0)[注 2]

注 **1.** 仅限μPD78F044x 和 78F045x。
2. 仅限μPD78F045x 和 78F046x 。

(1) 端口模式寄存器(PM1 至 PM4，PM8，PM10 至 PM12，PM14，PM15)

这些寄存器按位指定端口的输入或输出模式。

这些寄存器可以通过 1 位或 8 位存储器操作指令来设置。

复位信号的产生会将这些寄存器的内容设置为 FFH。

当端口引脚用作复用功能时，端口模式寄存器的设置请参考 **4.5 使用复用功能时，PFALL，PF2，PF1，ISC，端口模式寄存器和输出锁存的设置。**

图 **4-21.** 端口模式寄存器的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PM1	1	1	1	PM14	PM13	PM12	PM11	1	FF21H	FFH	R/W
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20	FF22H	FFH	R/W
PM3	1	1	1	PM34	PM33	PM32	PM31	1	FF23H	FFH	R/W
PM4	1	1	1	PM44	PM43	PM42	PM41	PM40	FF24H	FFH	R/W
PM8	1	1	1	1	PM83	PM82	PM81	PM80	FF28H	FFH	R/W
PM10	1	1	1	1	PM103	PM102	PM101	PM100	FF2AH	FFH	R/W
PM11	1	1	1	1	PM113	PM112	PM111	PM110	FF2BH	FFH	R/W
PM12	1	1	1	1	1	1	1	PM120	FF2CH	FFH	R/W
PM14	1	1	1	1	PM143	PM142	PM141	PM140	FF2EH	FFH	R/W
PM15	1	1	1	1	PM153	PM152	PM151	PM150	FF2FH	FFH	R/W

PMmn	Pmn 引脚输入/输出模式选择 (m = 1 至 4，8，10 至 12，14，15；n = 0 至 7)
0	输出模式 (输出缓冲器打开)
1	输入模式 (输出缓冲器关闭)

注意事项 请确保将 **PM1** 的第 **0** 位和第 **5** 位至第 **7** 位，**PM3** 的第 **0** 位和第 **5** 位至第 **7** 位，**PM4** 的第 **5** 位至第 **7** 位，**PM8** 的第 **4** 位至第 **7** 位，**PM10** 的第 **4** 位至第 **7** 位，**PM11** 的第 **4** 位至第 **7** 位，**PM12** 的第 **1** 位至第 **7** 位，**PM14** 的第 **4** 位至第 **7** 位，**PM15** 的第 **4** 位至第 **7** 位设置为 **"1"**。

(2) 端口寄存器(P1 至 P4，P8 ，P10 至 P12，P14，P15)

当数据从端口输出时，这些寄存器用于写入从芯片输出的数据。

如果在输入模式下读取数据，则读取的是引脚电平。如果在输出模式下读取，则读取的是输出锁存器的值。

这些寄存器可以通过 1 位或 8 位存储器操作指令来设置。

复位信号的产生会将这些寄存器清零(00H)。

图 4-22. 端口寄存器的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
P1	0	0	0	P14	P13	P12	P11	0	FF01H	00H (输出锁存)	R/W
P2	P27	P26	P25	P24	P23	P22	P21	P20	FF02H	00H (输出锁存)	R/W
P3	0	0	0	P34	P33	P32	P31	0	FF03H	00H (输出锁存)	R/W
P4	0	0	0	P44	P43	P42	P41	P40	FF04H	00H (输出锁存)	R/W
P8	0	0	0	0	P83	P82	P81	P80	FF08H	00H (输出锁存)	R/W
P10	0	0	0	0	P103	P102	P101	P100	FF0AH	00H (输出锁存)	R/W
P11	0	0	0	0	P113	P112	P111	P110	FF0BH	00H (输出锁存)	R/W
P12	0	0	0	P124注 2	P123注 2	P122注 2	P121注 2	P120	FF0CH	00H注 1 (输出锁存)	R/W注 1
<R> P14	PK143注 3	PK142注 3	PK141注 3	PK140注 3	P143	P142	P141	P140	FF0EH	00H (输出锁存)	R/W
<R> P15	PK153注 3	PK152注 3	PK151注 3	PK150注 3	P153	P152	P151	P150	FF0FH	00H (输出锁存)	R/W

Pmn	m = 1 至 4，8，10 至 12，14，15；n = 0 至 7	
	输出数据控制(输出模式下)	输入数据读取 (输入模式下)
0	输出 0	输入低电平
1	输出 1	输入低电平

注 1. P121 至 P124 是只读的。 在复位时，这些变为不确定。

2. 当引脚的操作模式是时钟输入模式，总是读到 0。

<R> 3. 该位用于 segment 键扫描功能。详情参见 **17.3 控制 LCD 控制器/驱动器的寄存器**。

(3) 上拉电阻选择寄存器**(PU1，PU3，PU4，PU8，PU10** 至 **PU12，PU14，PU15)**

这些寄存器指定是否使用 P11 至 P14，P31 至 P34，P40 至 P44，P80 至 P83，P100 至 P103，P110 至 P113，P120，P140 至 P143 或 P150 至 P153 的内部上拉电阻。只有那些设置为输入模式且已通过 PU1，PU3，PU4，PU8，PU10 至 PU12，PU14，PU15 指定使用内部上拉电阻的引脚，才可以使用内部上拉电阻。不论如何设置 PU1，PU3，PU4，PU8，PU10 至 PU12，PU14，PU15，设置为输出模式的引脚以及用作复用功能的输出引脚都无法连接内部上拉电阻。

这些寄存器可以通过 1 位或 8 位存储器操作指令来设置。

复位信号的产生会将这些寄存器清零(00H)。

图 **4-23.** 上拉电阻选择寄存器的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PU1	0	0	0	PU14	PU13	PU12	PU11	0	FF31H	00H	R/W
PU3	0	0	0	PU34	PU33	PU32	PU31	0	FF33H	00H	R/W
PU4	0	0	0	PU44注	PU43注	PU42注	PU41注	PU40注	FF34H	00H	R/W
PU8	0	0	0	0	PU83	PU82	PU81	PU80	FF38H	00H	R/W
PU10	0	0	0	0	PU103	PU102	PU101	PU100	FF3AH	00H	R/W
PU11	0	0	0	0	PU113	PU112	PU111	PU110	FF3BH	00H	R/W
PU12	0	0	0	0	0	0	0	PU120	FF3CH	00H	R/W
PU14	0	0	0	0	PU143	PU142	PU141	PU140	FF3EH	00H	R/W
PU15	0	0	0	0	PU153	PU152	PU151	PU150	FF3FH	00H	R/W

PUmn	Pmn 引脚内部上拉电阻选择 (m = 1 至 4，8，10 至 12，14，15；n = 0 至 7)
0	不连接内部上拉电阻
1	连接内部上拉电阻

<R> 注 该位用于 segment 键扫描功能。详情参见 **17.3** 控制 **LCD** 控制器**/**驱动器的寄存器。

(4) 端口功能寄存器 1 (PF1)

该寄存器用于设置 P13/SO10/TxD0/<TxD6>引脚的功能。

可以通过 1 位或 8 位存储器操作指令来设置 PF1。

复位信号的产生会将 PF1 清零（00H）。

图 4-24. 端口功能寄存器 1 (PF1)的格式

地址：FF20H 复位后 00H R/W

符号	7	6	5	4	3	2	1	0
PF1	0	0	0	0	PF13	0	0	0

PF13	端口 (P13)，CSI10，UART0和 UART6 输出规范
0	用作 P13 或 SO10
1	用作 TxD0或TxD6

(5) 端口功能寄存器 2 (PF2) (仅限μPD78F044x 和 78F045x)

该寄存器用于设置 P20 至 P27 引脚作为端口引脚（除 segment 输出引脚外）或 segment 输出引脚。

可以通过 1 位或 8 位存储器操作指令来设置 PF2。

复位信号的产生会将 PF2 清零（00H）。

图 4-25. 端口功能寄存器 2 (PF2)的格式

地址：FFB5H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
PF2	PF27	PF26	PF25	PF24	PF23	PF22	PF21	PF20

PF2n	端口/segment 输出规范
0	用作端口(除segment 输出外)
1	用作segment 输出

备注 n = 0 至 7

(6) 端口功能寄存器 ALL (PFALL)

该寄存器用于设置 P8，P10，P11，P14 和 P15 引脚作为端口引脚（除 segment 输出引脚外）或 segment 输出引脚。

可以通过 1 位或 8 位存储器操作指令来设置 PFALL。

复位信号的产生会将 PFALL 清零（00H）。

图 4-26. 端口功能寄存器 ALL (PFALL)的格式

地址：FFB6H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
PFALL	0	PF15ALL	PF14ALL	0	PF11ALL	PF10ALL	0	PF08ALL

PFnALL	端口/segment 输出规范
0	用作端口(除segment 输出外)
1	用作segment 输出

备注 n = 08，10，11，14，15

(7) A/D 端口配置寄存器 0 (ADPC0) (仅限μPD78F045x 和 78F046x)

该寄存器将 P20/ANI0 至 P27/ANI7 引脚切换为 A/D 转换器的模拟输入或数字输入/输出端口。

可以通过 1 位或 8 位存储器操作指令来设置 ADPC0。

复位信号的产生会将该寄存器置为 08H。

图 4-27. A/D 端口配置寄存器 0 (ADPC0)的格式

地址: FF8FH 复位后: 08H R/W

符号	7	6	5	4	3	2	1	0
ADPC0	0	0	0	0	ADPC03	ADPC02	ADPC01	ADPC00

ADPC03	ADPC02	ADPC01	ADPC00	数字 I/O (D)/模拟输入 (A: 逐次逼近型, Δ: ΔΣ 型) 切换							
				P27/ANI7/REF+	P26/ANI6/REF-	P25/ANI5/DS2+	P24/ANI4/DS2-	P23/ANI3/DS1+	P22/ANI2/DS1-	P21/ANI1/DS0+	P20/ANI0/DS0-
0	0	0	0	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ
0	0	0	1	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A	D
0	0	1	0	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	D	D
0	0	1	1	A/Δ	A/Δ	A/Δ	A/Δ	A	D	D	D
0	1	0	0	A/Δ	A/Δ	A/Δ	A/Δ	D	D	D	D
0	1	0	1	A	A	A	D	D	D	D	D
0	1	1	0	A	A	D	D	D	D	D	D
0	1	1	1	A	D	D	D	D	D	D	D
1	0	0	0	D	D	D	D	D	D	D	D
其它情况				禁止设置							

注意事项 **1.** 使用端口模式寄存器 **2(PM2)**将用于 **A/D** 转换的通道设置为输入模式。

2. 通过 **ADPC0** 而不是 **ADS**，**ADDS1** 或 **ADDS0** 将该引脚设置为数字输入**/**输出，

3. 如果将数据写入 **ADPC0**，产生一个等待周期。当 **CPU** 运行于副系统时钟且外设硬件时钟停止时，不要将数据写入 **ADPC0**。详情参见 第三十三章 等待注意事项。

4. 如果 **ANI0/P20/SEG31** 至 **ANI7/P27/SEG24** 引脚由 **PF2** 设置为 **segment** 输出，不论 **ADPC0** 如何设置**(**仅限μ**PD78F045x)**，输出都被设置为 **segment** 输出。

4.4 端口功能操作

设置为输入模式和输出模式的端口操作是不同的，具体如下所示。

注意事项 在 1 位操作指令的情况下，虽然只操作了一个信号位，该端口仍被作为 8 位单元来访问。因此，对于一个输入输出引脚混合的端口，指定为输入状态的引脚，即使不是被操作的位，其输出锁存器的内容不确定。

4.4.1 写入输入/输出端口

(1) 输出模式

使用传输指令对输出锁存器进行写操作，输出锁存器的内容从引脚输出。
一旦数据被写入输出锁存器，它将一直保持，直到再次写入数据。
复位信号产生时，输出锁存器的内容被清零。

(2) 输入模式

使用传输指令将一个值写入输出锁存器，因为输出缓冲器处于关闭状态，所以引脚状态不会改变。
一旦数据被写入输出锁存器，它将一直保持，直到再次写入数据。

4.4.2 读取输入/输出端口

(1) 输出模式

使用传输指令读取到的是输出锁存器的内容。输出锁存器的内容不会改变。

(2) 输入模式

使用传输指令读取到的是引脚状态。输出锁存器的内容不会改变。

4.4.3 输入/输出端口的操作

(1) 输出模式

对输出锁存器的内容执行一个操作时，操作结果写入输出锁存器。而输出锁存器的内容从引脚输出。
一旦数据被写入输出锁存器，它将一直保持，直到再次写入数据。
复位信号产生时，输出锁存器的内容被清零。

(2) 输入模式

读取引脚电平，且对它的内容执行操作，操作的结果写入输出锁存器，因为输出缓冲器处于关闭状态，所以引脚状态不会改变。

4.5 使用复用功能时，PFALL，PF2，PF1，ISC端口模式寄存器和输出锁存器的设置

要使用端口引脚的复用功能，应按照表 4-5 对 PFALL，PF2，PF1，ISC，端口模式寄存器和输出锁存器进行设置。

表 **4-5.** 使用复用功能时，**PFALL**，**PF2**，**PF1**，**ISC** 端口模式寄存器和输出锁存器的设置**(1/2)**

引脚名称	复用功能		PFALL, PF2[注 4]	PF1	ISC	PM××	P××
	功能名称	I/O					
P11	$\overline{SCK10}$	输入	–			1	×
		输出	–			0	1
P12	SI10	输入	–			1	×
	RxD0	输入	–			1	×
	<RxD6>	输入	–		ISC4 = 0, ISC5 = 1[注 5, 7]	1	×
P13[注 10]	SO10	输出	–	PF13 = 0		0	0
	TxD0	输出	–	PF13 = 1		0	×
	<TxD6>	输出	–	PF13 = 1	ISC4 = 0, ISC5 = 1	0	×
P14	INTP4	输入	–			1	×
P20 至 P27[注 2]	SEG31 至 SEG24[注 11]	输出	1			×	×
	ANI0 至 ANI7[注 1]	输入	0			1	×
	DS0± 至 DS2±[注 8]	输入	0			1	×
	REF±[注 8]	输入	0			1	×
P31	TOH1	输出	–			0	0
	INTP3	输入	–			1	×
P32	TOH0	输出	–			0	0
	MCGO	输出	–			0	0
P33	TI000	输入	–		ISC1 = 0	1	×
	RTCDIV	输出	–			0	0
	RTCCL	输出	–			0	0
	BUZ	输出	–			0	0
	INTP2	输入	–			1	×
P34	TI52	输入	–		注 **6**	1	×
	TI010	输入	–			1	×
	TO00	输出	–			0	0
	RTC1HZ	输出	–			0	0
	INTP1	输入	–			1	×

(注 和 备注 在下页列出)

表 **4-5.** 使用复用功能时，**PFALL**，**PF2**，**PF1**，**ISC** 端口模式寄存器和输出锁存器的设置 **(2/2)**

引脚名称	复用功能		PFALL, PF2[注 4]	ISC	PM××	P××
	功能名称	I/O				
P40	KR0	输入	–		1	×
	V_{LC3}[注 9]	输入	–		×	×
P41	KR1	输入	–		1	×
	RIN	输入	–		1	×
P42	KR2	输入	–		1	×
P43	KR3	输入	–		1	×
	TI51	输入	–		1	×
	TO51	输出	–		0	0
P44	KR4	输入	–		1	×
	TI50	输入	–		1	×
	TO50	输出	–		0	0
P80 至 P83	SEG4 至 SEG7	输出	1		×	×
P100 至 P103	SEG8 至 SEG11	输出	1		×	×
P110	SEG12	输出	1	ISC3 = 0	×	×
P111	SEG13	输出	1	ISC3 = 0	×	×
P112	SEG14	输出	1	ISC3 = 0	×	×
	TxD6	输出	0	ISC3 = 1, ISC4 = ISC5 = 0	0	1
P113	SEG15	输出	1	ISC3 = 0	×	×
	RxD6	输入	0	ISC3 = 1, ISC4 = ISC5 = 0[注 5, 7]	1	×
P120	EXLVI	输入	–		1	×
	INTP0	输入	–	ISC0 = 0	1	×
P121	X1[注 3]	–	–		×	×
	OCD0A	–	–		×	×
P122	X2[注 3]	–	–		×	×
	EXCLK[注 3]	输入	–		×	×
	OCD0B	–	–		×	×
P123	XT1[注 3]	–	–		×	×
P124	XT2[注 3]	–	–		×	×
P140 至 P143	SEG16 (KS0) 至 SEG19 (KS3)	输出	1		×	×
P150 至 P153	SEG20 (KS4) 至 SEG23 (KS7)	输出	1		×	×

（注和备注在下页列出）

注 **1.** 仅限μPD78F045x 和 78F046x 。

2. 根据端口功能寄存器 2 (PF2)、A/D 端口配置寄存器 0 (ADPC0) 、端口模式寄存器 2 (PM2)、模拟输入通道规格寄存器(ADS)和 ΔΣ A/D 转换器模式寄存器 0 (ADDCTL0)的设置，可以选择 P20/ANI0/DS0–，P21/ANI1/DS0+ ， P22/ANI2/DS1– ， P23/ANI3/DS1+ ， P24/ANI4/DS2– ， P25/ANI5/DS2+ ，P26/ANI6/REF–，和 P27/ANI7/REF+引脚的功能。

表 4-6. 设置 P20/SEG39 [注1]**/ANI0** [注2]**/DS0–**[注3] **至 P27/SEG32** [注1]**/ANI7** [注2]**/REF+**[注3] **引脚的功能设置**

PF2 [注1]	ADPC0	PM2	ADS	ADDCTL0	P20/SEG39 [注1]/ANI0 [注2]/DS0–[注3] 至 P27/SEG32 [注1]/ANI7 [注2]/REF+[注3] 引脚
数字/模拟选择	模拟输入选择	输入模式	不选择 ANI	不选择 DSn±	模拟输入 (不被转换)
			选择 ANI	不选择 DSn±	模拟输入 (可由逐次逼近型 A/D 转换器进行转换)
			不选择 ANI	选择 DSn±	模拟输入(可由 16 位ΔΣ型 A/D 转换器进行转换))
			选择 ANI	选择 DSn±	禁止设置
		输出模式	–		禁止设置
	数字输入/输出选择	输入模式	–		数字输入
		输出模式	–		数字输出
SEG 输出选择[注1]	–	–	–		Segment 输出[注1]

注 **1.** 仅限μPD78F044x 和 78F045x 。

2. 仅限μPD78F045x 和 78F046x。

3. 仅限μPD78F046x。

备注 n = 0 至 2

3. 当 P121 至 P124 引脚用于连接主系统时钟（X1，X2）或副系统时钟（XT1，XT2）的振荡器时，或者输入外部时钟（EXCLK）作为主系统时钟时，必须使用时钟操作模式选择寄存器（OSCCTL）对 X1 振荡模式、XT1 振荡模式或者外部时钟输入模式进行设置（如需了解详细信息，参见 **5.3 (1) 时钟操作模式选择寄存器（OSCCTL）**和（**3**）**副系统时钟引脚操作模式设置**）。OSCCTL 的复位值为 00H（P121 至 P124 都用作输入端口引脚）。

4. 每个端口对应的目标寄存器。

5. 设置 ISC1 = 1，RxD6 被设置用作 TI000 的输入源。

6. 设置 ISC2 = 1，可以通过 TMH2 来控制 TM52 的输入使能。

7. 设置 ISC0 = 1，RxD6 被设置用作 INTP0 的输入源。

8. 仅限 μPD78F046x。

9. 当 P40/KR0/V_{LC3} 引脚被设置为 1/4 偏压方式时，它被用作 V_{LC3}。当它被设置为其它偏压方式时，可以用作端口功能(P40)或按键中断功能(KR0)。

10. 用作端口功能时，设置 PF13 = 0 。

11. 用作端口功能时，设置 PF16 = 0 。

12. 仅限μPD78F044x 和 78F045x。

备注 **1.** ×： 无须理会

–： 不适用

PM××：端口模式寄存器

P××： 端口输出锁存

2. 尖括号 (< >) 内的功能由输入转换控制寄存器 (ISC)分配。

3. 当使用片上调试功能时，X1、X2 引脚可以用作片上调试模式设置引脚 (OCD1A，OCD1B) 。详情参见**第二十八章 片上调试功能**

第五章 时钟发生器

5.1 时钟发生器的功能

时钟发生器用于产生时钟，并提供给 CPU 和外设硬件设备。
可以选择以下三种系统时钟和时钟振荡器。

（1） 主系统时钟

<1> X1 振荡器

连接一个振荡器到 X1 和 X2，该振荡电路产生 f_X =2 至 10MHz 的时钟。
通过执行 STOP 指令或设置主 OSC 控制寄存器（MOC），可以停止振荡。

<2> 内部高速振荡器

这个振荡电路产生一个 f_{RH} = 8 MHz（典型值）的时钟。复位释放后，CPU 总是用这个内部高速振荡时钟启动操作。可以通过执行 STOP 指令或者使用内部振荡模式寄存器（RCM）停止其振荡。

外部主系统时钟（f_{EXCLK} = 2 至 10MHz）也可以通过 OCD0B/EXCLK/X2/P122 引脚提供。可以通过执行 STOP 指令或者使用 RCM 禁止外部主系统时钟输入。
可以通过主时钟模式寄存器（MCM）选择内部高速系统时钟（X1 时钟或者外部主系统时钟）或内部高速振荡时钟，作为主系统时钟。

（2） 副系统时钟

- **副系统时钟振荡器**

在 XT1 和 XT2 之间连接一个 32.768kHz 的振荡器，该电路以 f_{XT} = 32.768 kHz 的频率进行振荡。通过处理器时钟控制寄存器（PCC）和时钟操作模式选择寄存器（OSCCTL），可以停止振荡。

备注 **1.** f_X： X1 时钟振荡频率
2. f_{RH}： 内部高速振荡时钟频率
3. f_{EXCLK}： 外部主系统时钟频率
4. f_{XT}： XT1 时钟振荡频率

（3） 内部低速振荡时钟（看门狗定时器的时钟）

- 内部低速振荡器

该电路以 f_{RL} = 240 kHz（典型值）的时钟振荡。复位释放后，内部低速振荡时钟总是开始操作。
通过选项字节设置"内部低速振荡器可由软件停止"时，可以通过内部振荡模式寄存器（RCM）停止其振荡。
内部低速振荡时钟不能用作 CPU 时钟。如下硬件使用内部低速振荡时钟。

- 看门狗定时器
- 8 位定时器 H1（如果选择 f_{RL}，$f_{RL}/2^7$ 或 $f_{RL}/2^9$ 作为计数时钟）
- LCD 控制器/驱动器 （如果选 $f_{RL}/2^3$ 为 LCD 源时钟）

备注 f_{RL}： 内部低速振荡时钟频率

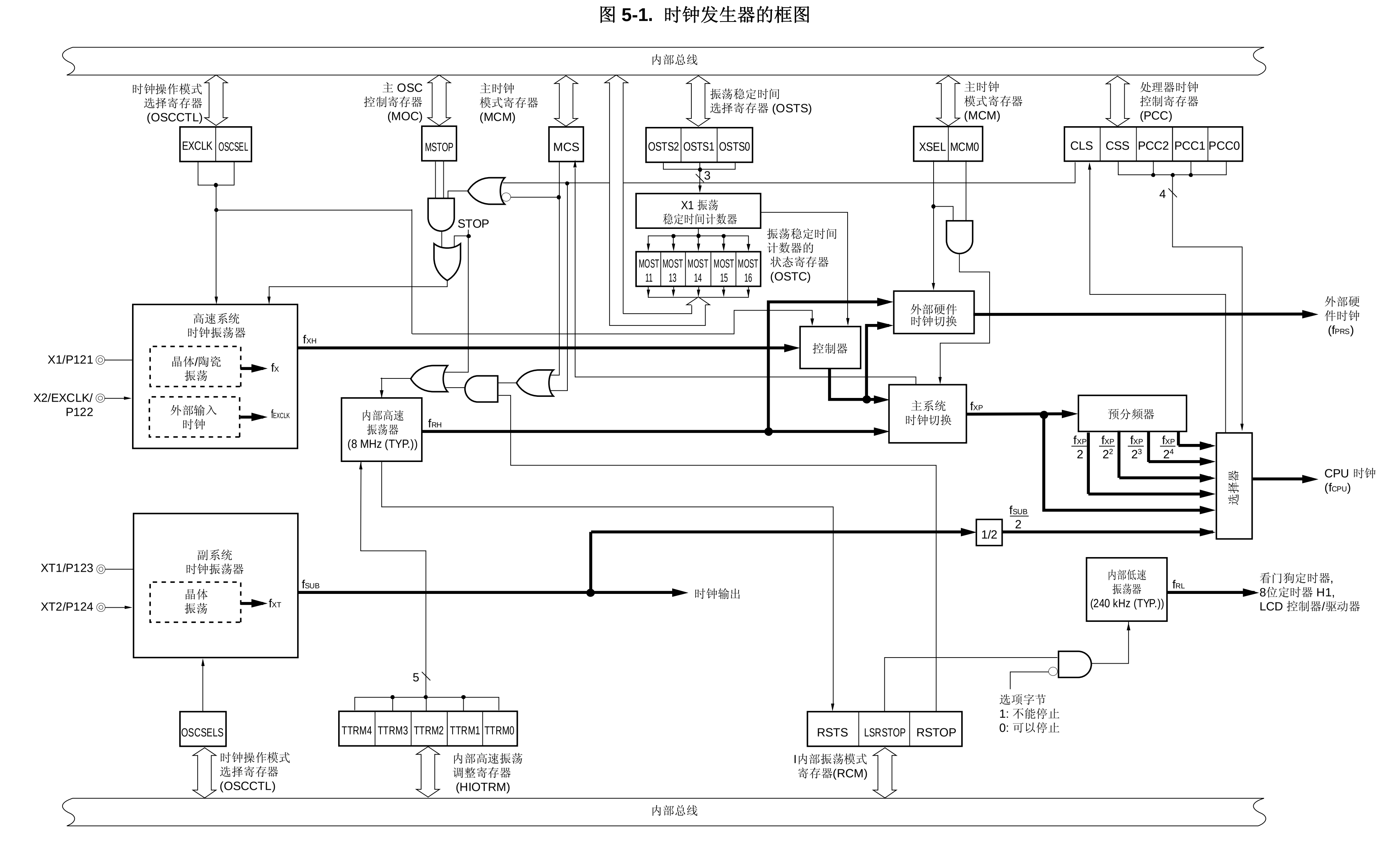
5.2 时钟发生器的配置

时钟发生器包括以下硬件。

表 5-1. 时钟发生器的配置

项目	构成
控制寄存器	时钟操作模式选择寄存器（OSCCTL） 处理器时钟控制寄存器 （PCC） 内部振荡模式寄存器 （RCM） 主 OSC 控制寄存器 （MOC） 主时钟模式寄存器 （MCM） 振荡稳定时间计数器状态寄存器（OSTC） 振荡稳定时间选择寄存器（OSTS） 内部高速振荡微调寄存器（HIOTRM）
振荡器	X1 振荡器 XT1 振荡器 内部高速振荡器 内部低速振荡器

图 5-1. 时钟发生器的框图



备注
1. f_X： X1 时钟振荡频率
2. f_{RH}： 内部高速振荡时钟频率
3. f_{EXCLK}： 外部主系统时钟频率
4. f_{XH}： 高速系统时钟频率
5. f_{XP}： 主系统时钟频率
6. f_{PRS}： 外设硬件时钟频率
7. f_{CPU}： CPU 时钟频率
8. f_{XT}： XT1 时钟振荡频率
9. f_{SUB}： 副系统时钟频率
10. f_{RL}： 内部低速振荡时钟频率

5.3 控制时钟发生器的寄存器

以下八个寄存器用于控制时钟发生器。

- 时钟操作模式选择寄存器 （OSCCTL）
- 处理器时钟控制寄存器 （PCC）
- 内部振荡模式寄存器 （RCM）
- 主 OSC 控制寄存器（MOC）
- 主时钟模式寄存器 （MCM）
- 振荡稳定时间计数器状态寄存器 （OSTC）
- 振荡稳定时间选择寄存器 （OSTS）
- 内部高速振荡微调寄存器 （HIOTRM）

（1） 时钟操作模式选择寄存器 （OSCCTL）

该寄存器用于选择高速系统时钟和副系统时钟的操作模式，以及内置振荡器的获取方式。
可以通过 1 位或 8 位存储器操作指令来设置 OSCCTL。
复位信号的产生会将该寄存器清除为 00H。

图 **5-2.** 时钟操作模式选择寄存器的格式（**OSCCTL**）

地址： FF9FH　复位后： 00H　R/W

符号	<7>	<6>	5	<4>	3	2	1	0
OSCCTL	EXCLK	OSCSEL	0	OSCSELS	0	0	0	0

EXCLK	OSCSEL	高速系统时钟引脚操作模式	P121/X1 引脚	P122/X2/EXCLK 引脚
0	0	输入端口模式	输入端口	
0	1	X1 振荡模式	晶体/陶瓷振荡器连接	
1	0	输入端口模式	输入端口	
1	1	外部时钟输入模式	输入端口	外部时钟输入

注意事项　要修改 **EXCLK** 和 **OSCSEL** 的值，必须确保主 **OSC** 控制寄存器（**MOC**）的第 **7** 位（**MSTOP**）**=1**（**X1** 振荡器停止或禁止使用从 **EXCLK** 引脚输入的外部时钟）。
请确保将第 **0** 位至第 **3** 位和第 **5** 位清除为**“0”**。

备注　f_{XH}：高速系统时钟振荡频率

（2） 处理器时钟控制寄存器（PCC）

该寄存器用于选择 CPU 时钟、分频比和副系统时钟的操作模式。

可以通过 1 位或 8 位存储器操作指令来设置 PCC。

复位信号的产生会将 PCC 设置为 01H。

图 5-3. 处理器时钟控制寄存器（PCC）的格式

地址： FFFBH 复位后： 01H R/W 注 1

符号	7	6	<5>	<4>	3	2	1	0
PCC	0	0	CLS	CSS	0	PCC2	PCC1	PCC0

CLS	CPU 时钟状态
0	主系统时钟
1	副系统时钟

CSS	PCC2	PCC1	PCC0	CPU 时钟 （f_{CPU}） 选择
0	0	0	0	f_{XP}
	0	0	1	$f_{XP}/2$ （默认）
	0	1	0	$f_{XP}/2^2$
	0	1	1	$f_{XP}/2^3$
	1	0	0	$f_{XP}/2^4$
1	0	0	0	$f_{SUB}/2$
	0	0	1	
	0	1	0	
	0	1	1	
	1	0	0	
其他情况				禁止设置

注 第 5 位是只读的。

注意事项 **请确保将第 3 位，第 6 位和第 7 位清除为“0”。**

备注 **1.** f_{XP}： 主系统时钟振荡频率

2. f_{SUB}： 副系统时钟振荡频率

在 78K0/LE3 中执行速度最快的指令执行时间在 2 个 CPU 时钟以内。CPU 时钟（f_{CPU}）与指令最短执行时间的关系如表 5-2 所示。

表 **5-2. CPU** 时钟与指令最短执行时间的关系

CPU 时钟（f_{CPU}）	指令最短执行时间： 2/f_{CPU}		
	主系统时钟		副系统时钟
	高速系统时钟[注]	内部高速振荡时钟[注]	
	运行于 10 MHz	运行于 8 MHz（典型值）	运行于 32.768 kHz
f_{XP}	0.2 μs	0.25 μs（典型值）	–
$f_{XP}/2$	0.4 μs	0.5 μs （典型值）	–
$f_{XP}/2^2$	0.8 μs	1.0 μs （典型值）	–
$f_{XP}/2^3$	1.6 μs	2.0 μs （典型值）	–
$f_{XP}/2^4$	3.2 μs	4.0 μs （典型值）	–
$f_{SUB}/2$	–	–	122.1 μs

注　主时钟模式寄存器（MCM）用于设置提供给 CPU 的主系统时钟（高速系统时钟/ 内部高速振荡时钟）（参见图 **5-6**）。

（3）　副系统时钟引脚操作模式的设置

使用时钟操作模式选择寄存器（OSCCTL）的第 4 位（OSCSELS）来设置副系统时钟引脚操作模式。

表 **5-3.** 副系统时钟引脚操作模式的设置

OSCCTL 第 4 位	副系统时钟引脚操作模式	P123/XT1 引脚	P124/XT2 引脚
OSCSELS			
0	输入端口模式	输入端口	
1	XT1 振荡模式	晶体振荡器连接	

注意事项　**改变 OSCSELS 的当前值时，请确保处理器时钟控制寄存器（PCC）的第 5 位（CLS）= 0 （CPU 运行于主系统时钟）。**

（4）　内部振荡模式寄存器 （RCM）

该寄存器用于设置内部振荡器的操作模式。

可以通过 1 位或 8 位存储器操作指令来设置 RCM。

复位信号的产生会将该寄存器设置为 80H[注 1]。

图 5-4. 内部振荡模式寄存器（RCM）的格式

地址： FFA0H　复位后： 80H[注 1]　R/W[注 2]

符号	<7>	6	5	4	3	2	<1>	<0>
RCM	RSTS	0	0	0	0	0	LSRSTOP	RSTOP

RSTS	内部高速振荡器的状态
0	等待内部高速振荡器的精确稳定
1	内部高速振荡器的稳定操作

LSRSTOP	内部低速振荡器振荡/停止
0	内部低速振荡器振荡
1	内部低速振荡器停止

RSTOP	内部高速振荡器振荡/停止
0	内部高速振荡器振荡
1	内部高速振荡器停止

注　**1.** 复位释放后，该寄存器的值立即变为 00H，但在内部高速振荡器已经稳定后自动变为 80H。

　　2. 第 7 位是只读的。

注意事项　当设置 RSTOP = 1 时，请务必确保 CPU 使用的时钟不是内部高速振荡时钟。具体地说，在下列任一条件下进行设置。

- **当 MCS = 1 （当 CPU 运行于高速系统时钟时）**
- **当 CLS = 1 （当 CPU 运行于副系统时钟时）**

此外，将 RSTOP 设置为 1 之前，应该停止正在使用内部高速振荡时钟的外设硬件操作。

（5）　主 OSC 控制寄存器（MOC）

该寄存器用于选择高速系统时钟的操作模式。

当 CPU 使用的时钟不是高速系统时钟时，该寄存器用来停止 X1 振荡器或禁止从 EXCLK 引脚输入外部时钟。

可以通过 1 位或 8 位存储器操作指令来设置 MOC。

复位信号的产生会将该寄存器设置为 80H。

图 5-5. 主 OSC 控制寄存器 （MOC）的格式

地址： FFA2H　复位后： 80H　R/W

符号	<7>	6	5	4	3	2	1	0
MOC	MSTOP	0	0	0	0	0	0	0

MSTOP	高速系统时钟操作的控制	
	X1 振荡模式	外部时钟输入模式
0	X1 振荡器操作	允许使用来自 EXCLK 引脚的外部时钟
1	X1 振荡器停止	禁止使用来自 EXCLK 引脚的外部时钟

注意事项

1. 当设置 **MSTOP = 1** 时，请务必确保 **CPU** 使用的时钟不是高速系统时钟。具体地说，在下列任一条件下进行设置。

- 当 **MCS = 0** （当 **CPU** 运行于内部高速振荡时钟时）
- 当 **CLS = 1** （当 **CPU** 运行于副系统时钟时）

此外，将 **MSTOP** 设置为 **1** 之前，应该停止正在使用高速系统时钟的外设硬件操作。

2. 当时钟操作模式选择寄存器（**OSCCTL**）的第 **6** 位（**OSCSEL**）**= 0**（**I/O** 端口模式）时，不要将 **MSTOP** 清除为 **0**。

3. 外设硬件时钟停止时，外设硬件不能操作。在外设硬件时钟停止后，要恢复外设硬件的操作，则必须对外设硬件进行初始化。

（6） 主时钟模式寄存器（MCM）

该寄存器用于选择提供给 CPU 的主系统时钟和提供给外设硬件的时钟。

可以通过 1 位或 8 位存储器操作指令来设置 MCM。

复位信号的产生会将该寄存器清除为（00H）。

图 5-6. 主时钟模式寄存器 （MCM）的格式

地址： FFA1H 复位后： 00H R/W 注

符号	7	6	5	4	3	<2>	<1>	<0>
MCM	0	0	0	0	0	XSEL	MCS	MCM0

<table>
<tr><th rowspan="2">XSEL</th><th rowspan="2">MCM0</th><th colspan="2">提供给主系统时钟和外设硬件的时钟选择</th></tr>
<tr><th>主系统时钟 （f_{XP}）</th><th>外设硬件时钟 （f_{PRS}）</th></tr>
<tr><td>0</td><td>0</td><td rowspan="3">内部高速振荡时钟（f_{RH}）</td><td rowspan="2">内部高速振荡时钟 （f_{RH}）</td></tr>
<tr><td>0</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td rowspan="2">高速系统时钟 （f_{XH}）</td></tr>
<tr><td>1</td><td>1</td><td>高速系统时钟 （f_{XH}）</td></tr>
</table>

MCS	主系统时钟状态
0	使用内部高速振荡时钟
1	使用高速系统时钟

注 第 1 位是只读的。

注意事项 **1.** 在复位释放后，**XSEL** 只能被修改一次。

2. 无论 **XSEL** 和 **MCM0** 如何设置，除 f_{PRS} 外的某个时钟将被提供给下列外设功能

- 看门狗定时器（使用内部低速振荡时钟）
- 选择"f_{RL}"，"$f_{RL}/2^7$"或者"$f_{RL}/2^9$"作为 **8** 位定时器 **H1** 的计数时钟（使用内部低速振荡时钟）
- 选择 "$f_{RL}/2^3$" 作为 **LCD** 控制器**/**驱动器的 **LCD** 源时钟（使用内部低速振荡时钟）
- 选择外部时钟作为外设硬件的时钟源
（以下情况除外：选择 **TM0n** （**n = 0，1**）的外部计数时钟（**TI00n** 有效边沿）时）

（7） 振荡稳定时间计数器状态寄存器 （OSTC）

该寄存器用于指示 X1 时钟振荡稳定时间计数器的计数状态。当 X1 时钟振荡启动且使用内部高速振荡时钟或副系统时钟作为 CPU 时钟，可以检查 X1 时钟振荡稳定时间。

可以通过 1 位或 8 位存储器操作指令来读取 OSTC。

复位释放后（由 $\overline{\text{RESET}}$ 输入、POC、LVI、和 WDT 进行复位），STOP 指令或设置 MSTOP（MOC 寄存器的第 7 位） = 1，都可以将 OSTC 清除为 00H。

图 5-7. 振荡稳定时间计数器状态寄存器（OSTC）的格式

地址： FFA3H 复位后： 00H R

符号	7	6	5	4	3	2	1	0
OSTC	0	0	0	MOST11	MOST13	MOST14	MOST15	MOST16

MOST 11	MOST 13	MOST 14	MOST 15	MOST 16	振荡稳定时间状态			
						f_X = 2 MHz	f_X = 5 MHz	f_X = 10 MHz
1	0	0	0	0	$2^{11}/f_X$ 最小值	1.02 ms 最小值	409.6 μs 最小值	204.8 μs 最小值
1	1	0	0	0	$2^{13}/f_X$ 最小值	4.10 ms 最小值	1.64 ms 最小值	819.2 μs 最小值
1	1	1	0	0	$2^{14}/f_X$ 最小值	8.19 ms 最小值	3.27 ms 最小值	1.64 ms 最小值
1	1	1	1	0	$2^{15}/f_X$ 最小值	16.38 ms 最小值	6.55 ms 最小值	3.27 ms 最小值
1	1	1	1	1	$2^{16}/f_X$ 最小值	32.77 ms 最小值	13.11 ms 最小值	6.55 ms 最小值

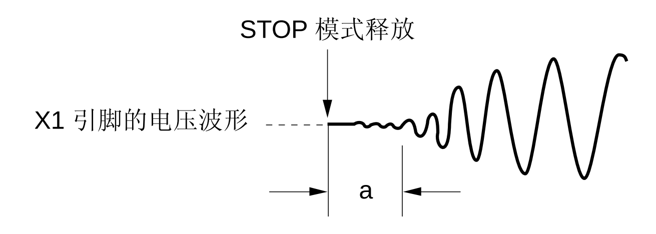
注意事项 **1.** 在经过上述时间后，从 **MOST11** 开始，逐位依次被设置为 **1**，并一直保持为 **1**。

2. 振荡稳定时间计数器向上计数到 **OSTS** 设置的振荡稳定时间。当内部高速振荡时钟作为 **CPU** 时钟时，如果进入 **STOP** 模式，则在释放该模式时，按如下方式设置振荡稳定时间。

- 预期的 **OSTC** 振荡稳定时间 ≤ **OSTS** 设置的振荡稳定时间。

因此要注意，在 **STOP** 模式释放后，仅在达到由 **OSTS** 设置的振荡稳定时间时，将状态赋给 **OSTC**。

3. **X1** 时钟振荡稳定等待时间不包括时钟振荡开始之前的时间（下图**“a”**表示的部分）。



备注 f_X： X1 时钟振荡频率

（8） 振荡稳定时间选择寄存器 （OSTS）

该寄存器用于选择当 STOP 模式释放时的 X1 时钟振荡稳定等待时间。

当 X1 时钟被选择作为 CPU 时钟时，在 STOP 模式释放后，操作需要等待的时钟由 OSTS 设置。

当内部高速振荡时钟被选择作为 CPU 时钟时，在 STOP 模式释放后，应通过 OSTC 来确认已经经过预期的振荡稳定时间。可以通过 OSTC 来检查是否达到设置的振荡稳定时间。

可以通过 8 位存储器操作指令来设置 OSTS。

复位信号的产生会将 OSTS 内容设置为 05H。

图 5-8. 振荡稳定时间选择寄存器（OSTS）的格式

地址： FFA4H 复位后： 05H R/W

符号	7	6	5	4	3	2	1	0
OSTS	0	0	0	0	0	OSTS2	OSTS1	OSTS0

OSTS2	OSTS1	OSTS0	振荡稳定时间的选择			
				f_X = 2 MHz	f_X = 5 MHz	f_X = 10 MHz
0	0	1	$2^{11}/f_X$	1.02 ms	409.6 μs	204.8 μs
0	1	0	$2^{13}/f_X$	4.10 ms	1.64 ms	819.2 μs
0	1	1	$2^{14}/f_X$	8.19 ms	3.27 ms	1.64 ms
1	0	0	$2^{15}/f_X$	16.38 ms	6.55 ms	3.27 ms
1	0	1	$2^{16}/f_X$	32.77 ms	13.11 ms	6.55 ms
其它情况			禁止设置			

注意事项 **1.** 当 **X1** 时钟被用作 **CPU** 时钟时，若要设置 **STOP** 模式，则在执行 **STOP** 指令之前设置 **OSTS**。

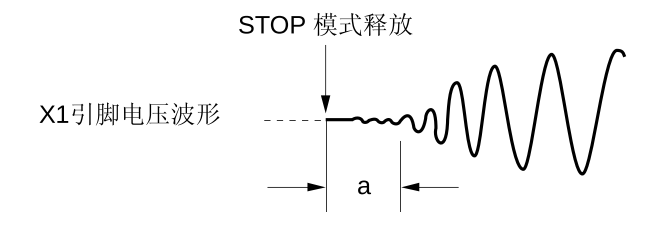
2. 在 **X1** 时钟振荡稳定时间内不要改变 **OSTS** 寄存器的值。

3. 振荡稳定时间计数器向上计数到 **OSTS** 设置的振荡稳定时间。当内部高速振荡时钟作为 **CPU** 时钟时，如果进入 **STOP** 模式，则在释放该模式时，按如下方式设置振荡稳定时间。

- 预期的 **OSTC** 振荡稳定时间 ≤ **OSTS** 设置的振荡稳定时间。

因此要注意，在 **STOP** 模式释放后，仅在达到由 **OSTS** 设置的振荡稳定时间时，将状态赋给 **OSTC**。

4. **X1** 时钟振荡稳定等待时间不包括时钟振荡开始之前的时间（下图**“a”**表示的部分）。



备注 f_X： X1 时钟振荡频率

（9）　内部高速振荡微调寄存器（HIOTRM）

该寄存器用于调整内部高速振荡器的精度。使用副系统时钟、晶体振荡器或使用高精度外部时钟输入的定时器，例如实时计数器，对内部高速振荡器频率自测量，对其精度进行修正，

可以通过 8 位存储器操作指令来设置 HIOTRM。

复位信号的产生会将 HIOTRM 设置为 10H。

注意事项　如果在精度调整后，温度或 V_{DD} 引脚电压发生变化，频率将会波动。同样，如果 HIOTRM 寄存器设置了除初值（10H）以外的值，由于随后发生的温度或 V_{DD} 电压变化，或因为 HIOTRM 设置值的改变，内部高速振荡时钟的振荡频率可能会超过 第三十章 电气特性（标准产品）中描述的最小值和最大值。如果温度或 V_{DD} 电压变动，在频率精度达到所需精度或稳定之前，必须执行频率精度修正。

<R>

图 5-9. 内部高速振荡微调寄存器（HIOTRM）的格式

地址： FF30H　复位后： 10H　R/W

符号	7	6	5	4	3	2	1	0
HIOTRM	0	0	0	TTRM4	TTRM3	TTRM2	TTRM1	TTRM0

TTRM4	TTRM3	TTRM2	TTRM1	TTRM0	时钟修正值（2.7 V ≤ V_{DD} ≤ 5.5 V）		
					最小值	典型值	最大值
0	0	0	0	0	–5.54%	–4.88%	–4.02%
0	0	0	0	1	–5.28%	–4.62%	–3.76%
0	0	0	1	0	–4.99%	–4.33%	–3.47%
0	0	0	1	1	–4.69%	–4.03%	–3.17%
0	0	1	0	0	–4.39%	–3.73%	–2.87%
0	0	1	0	1	–4.09%	–3.43%	–2.57%
0	0	1	1	0	–3.79%	–3.13%	–2.27%
0	0	1	1	1	–3.49%	–2.83%	–1.97%
0	1	0	0	0	–3.19%	–2.53%	–1.67%
0	1	0	0	1	–2.88%	–2.22%	–1.36%
0	1	0	1	0	–2.23%	–1.91%	–1.31%
0	1	0	1	1	–1.92%	–1.60%	–1.28%
0	1	1	0	0	–1.60%	–1.28%	–0.96%
0	1	1	0	1	–1.28%	–0.96%	–0.64%
0	1	1	1	0	–0.96%	–0.64%	–0.32%
0	1	1	1	1	–0.64%	–0.32%	±0%
1	0	0	0	0	±0% （默认）		
1	0	0	0	1	±0%	+0.32%	+0.64%
1	0	0	1	0	+0.33%	+0.65%	+0.97%
1	0	0	1	1	+0.66%	+0.98%	+1.30%
1	0	1	0	0	+0.99%	+1.31%	+1.63%
1	0	1	0	1	+1.32%	+1.64%	+1.96%
1	0	1	1	0	+1.38%	+1.98%	+2.30%
1	0	1	1	1	+1.46%	+2.32%	+2.98%
1	1	0	0	0	+1.80%	+2.66%	+3.32%
1	1	0	0	1	+2.14%	+3.00%	+3.66%
1	1	0	1	0	+2.48%	+3.34%	+4.00%
1	1	0	1	1	+2.83%	+3.69%	+4.35%
1	1	1	0	0	+3.18%	+4.04%	+4.70%
1	1	1	0	1	+3.53%	+4.39%	+5.05%
1	1	1	1	0	+3.88%	+4.74%	+5.40%
1	1	1	1	1	+4.24%	+5.10%	+5.76%

注意事项　**HIOTRM** 寄存器的指定值增加，内部高速振荡频率速度增快，如果指定值减小，则速度降低。相反的情况，比如指定值增加而频率速度降低，或指定值减小而速度增加，不会发生。

5.4 系统时钟振荡器

5.4.1 X1 振荡器

X1 振荡器通过连接到 X1 和 X2 引脚的晶体振荡器或陶瓷振荡器（2 至 10MHz）振荡。

也可以输入外部时钟。在这种情况下，从 EXCLK 引脚输入时钟信号。

图 5-10 是 X1 振荡器外部电路的示例。

图 5-10. X1 振荡器外部电路的示例



5.4.2 XT1 振荡器

XT1 振荡器通过连接到 XT1 和 XT2 引脚的晶体振荡器（标准值：32.768 kHz）振荡。

图 5-11 是 XT1 振荡器外部电路的示例。

图 5-11. XT1 振荡器外部电路的示例

（**a**） 晶体振荡

VSS
XT1
32.768
kHz
XT2

注意事项 **1.** **当使用 X1 振荡器和 XT1 振荡器时，图 5-10 和图 5-11 中虚线内区域应按照如下方法配线，以防止线间电容产生负面影响。**

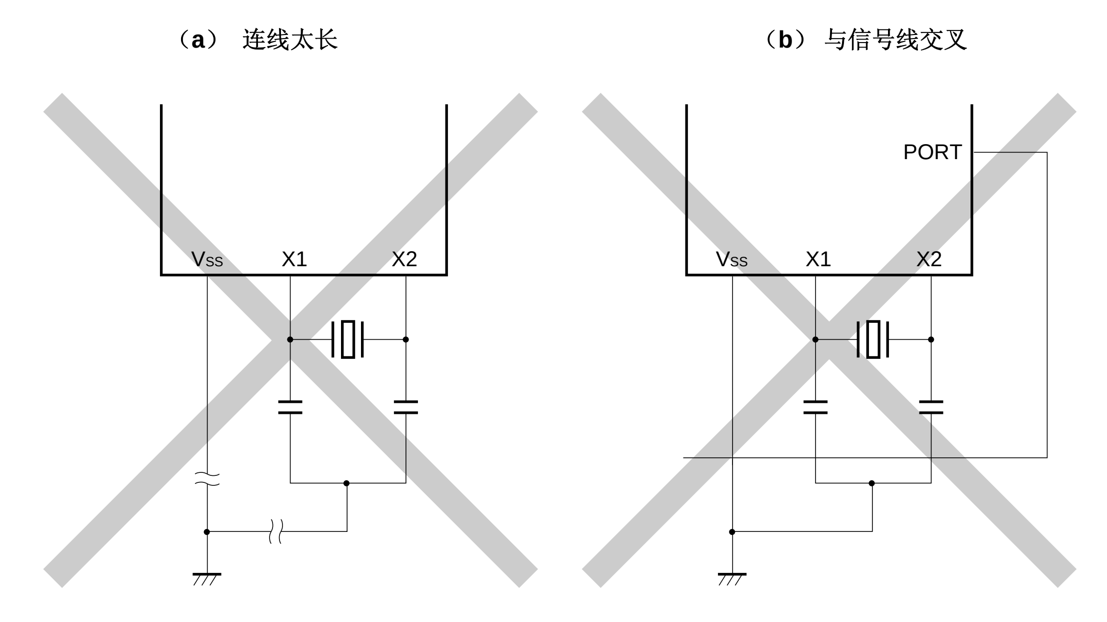
- **连接线越短越好。**
- **连接线不应与其他信号线交叉。不要将振荡器的布线放在流经电流变化较大的信号线周围。**
- **要保持振荡器电容器的接地点电位与 VSS 相同。不要将电容的地信号接入大电流通过的地模式。**
- **不要从振荡器拾取信号。**

请注意 XT1 振荡器被设计为低振幅电路，以降低功耗。

图 5-12 为不正确的谐振器连接示例。

图 5-12. 不正确的振荡器连接示例（**1/2**）

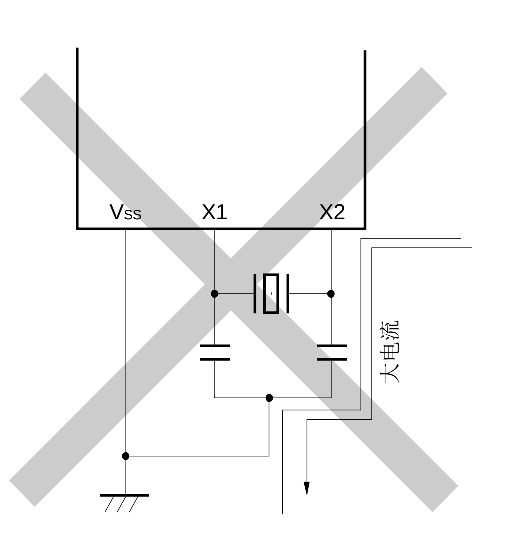
（**a**）　连线太长　　　　　　　　　　（**b**）　与信号线交叉



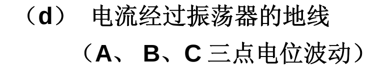
备注　　当使用副系统时钟时，分别用 XT1 和 XT2 代替 X1 和 X2。插入的串联电阻也连接在 XT2 侧。

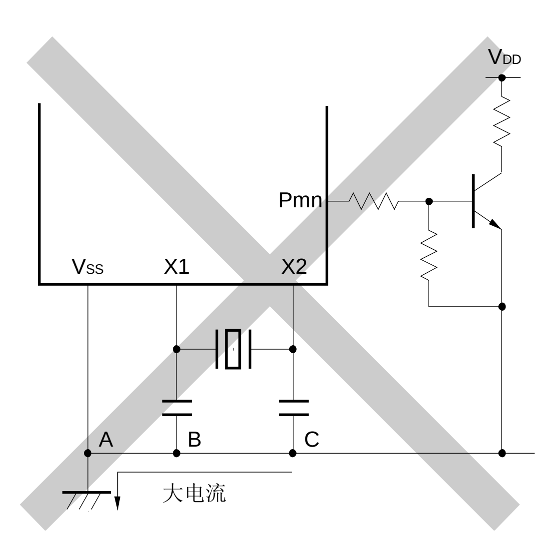
图 **5-12.** 不正确的振荡器连接示例（**2/2**）

（**c**） 连接线周围有大电流变化

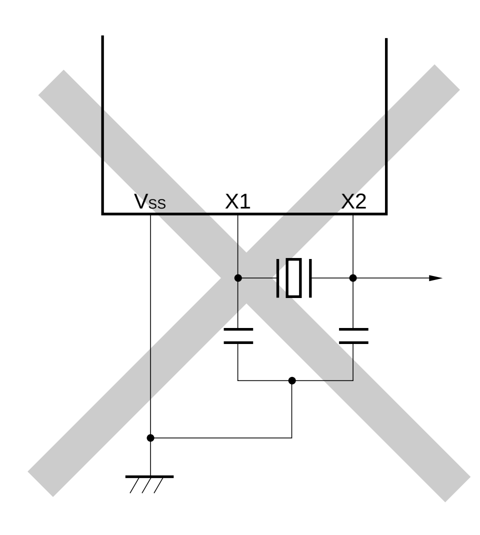


（**d**） 电流经过振荡器的地线


（**A**、 **B**、**C** 三点电位波动）



（**e**） 获取信号



备注　　在使用副系统时钟时，分别用 XT1 和 XT2 代替 X1 和 X2。插入的串联电阻也连接在 XT2 侧。

注意事项　当 X2 和 XT1 并行连接时，X2 的串扰噪音可能会叠加到 XT1，从而导致故障。

5.4.3 不使用副系统时钟时

如果无需使用副系统时钟来降低功耗，或者不使用副系统时钟作为 I/O 端口，则可以将 XT1 和 XT2 引脚设置为输入端口模式（OSCSELS = 0），并通过一个电阻单独连接到 V_{DD} 或 V_{SS}。

备注 OSCSELS： 时钟操作模式选择寄存器的第 4 位 （OSCCTL）

5.4.4 内部高速振荡器

78K0/LE3 产品中包含内部高速振荡器。可以通过内部振荡模式寄存器（RCM）控制振荡。

复位释放后，内部高速振荡器自动开始振荡（8 MHz （典型值））。

5.4.5 内部低速振荡器

78K0/LE3 产品中包含内部低速振荡器。

内部低速振荡时钟只作为看门狗定时器、8 位定时器 H1 和 LCD 控制器/驱动器的时钟使用。内部低速振荡时钟不能用作 CPU 时钟。

可以通过选项字节设置内部低速振荡器为“可由软件停止”或“不能停止”。如果设置“可由软件停止”，则可以通过内部振荡模式寄存器（RCM）控制振荡。

复位释放后，内部低速振荡器自动开始振荡，如果使用选项字节允许看门狗定时器操作，则可以驱动看门狗定时器（240 kHz （典型值））。

5.4.6 预分频器

当主系统时钟被选择作为提供给 CPU 的时钟时，预分频器对主系统时钟进行分频，可以产生多种时钟。

5.5 时钟发生器的操作

时钟发生器可以产生以下几种时钟，并控制 CPU 的操作模式，如待机模式。（参见图 **5-1**）。

- 主系统时钟 f_{XP}
  - 高速系统时钟 f_{XH}
    X1 时钟 f_X
    外部主系统时钟 f_{EXCLK}
  - 内部高速振荡时钟 f_{RH}
- 副系统时钟 f_{SUB}
  - XT1 时钟 f_{XT}
- 内部低速振荡时钟 f_{RL}
- CPU 时钟 f_{CPU}
- 外设硬件时钟 f_{PRS}

在 78K0/LE3 中，当复位释放后，当内部高速振荡器开始输出时，CPU 开始操作，因此具有以下特点。

（1） 安全功能的增强

如果 X1 时钟被默认设置为 CPU 时钟，在 X1 时钟遭到损坏或连接错误时设备不能操作，因此复位释放后也不能操作。但是，如果 CPU 的初始时钟是内部高速振荡时钟，则在复位释放后，可以由内部高速振荡时钟启动设备。于是，系统只需执行最少操作，比如由软件确认复位源或在出现故障时执行安全处理，便可以安全关闭系统。

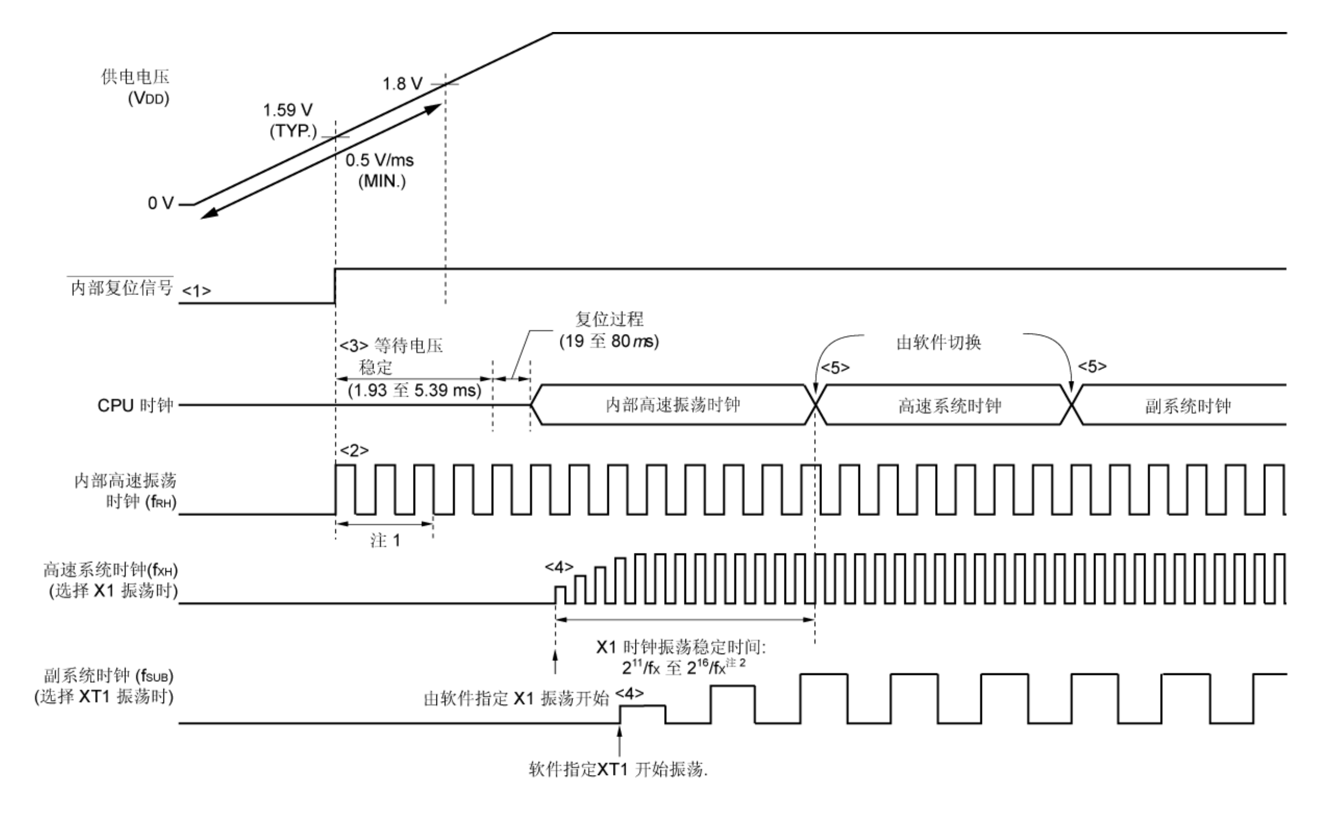
（2） 性能的改善

因为 CPU 无需等待 X1 时钟振荡稳定时间就可以启动，所以整体性能得到改善。

图 5-13 为打开供电电源上电时，时钟发生器的操作。

<R>

图 **5-13.** 打开供电电源上电时的时钟发生器操作
（设置为 **1.59 V POC** 模式 （选项字节：**POCMODE = 0**））



<1>　当电源打开时，通过上电清零 （POC） 电路产生一个内部复位信号。

<2>　当供电电压超过 1.59 V（典型值）时，复位被释放，且内部高速振荡器自动开始振荡。

<3>　当供电电压上升斜率为 0.5 V/ms（最大值）时，复位释放后且经过供电电压和稳压器的稳定时间后，CPU 开始运行于内部高速振荡时钟，然后执行复位处理。

<4>　通过软件设置 X1 或 XT1 时钟开始振荡（参见 **5.6.1 高速系统时钟控制示例**中的（**1**）和 **5.6.3 副系统时钟控制示例**中的（**1**））。

<5>　当 CPU 时钟切换到 X1 或 XT1 时钟时，等待时钟振荡稳定，然后通过软件设置切换（参见 **5.6.1 高速系统时钟控制示例**中的（**3**）和 **5.6.3 副系统时钟控制示例**中的（**3**））。

注　**1.** 内部电压稳定时间包括内部高速振荡时钟的振荡精确稳定时间。

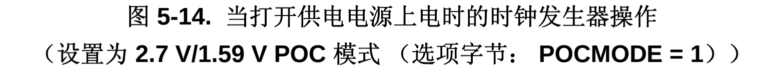
2. 复位释放时（如上图所示）或在 CPU 运行于内部高速振荡时钟时释放 STOP 模式，使用振荡稳定时间计数器状态寄存器（OSTC）来确认 X1 时钟的振荡稳定时间。如果 CPU 运行于高速系统时钟（X1 振荡），则可以使用振荡稳定时间选择寄存器（OSTS）来设置释放 STOP 模式时的振荡稳定时间。

注意事项　1. 在供电电压达到 1.8V 之前，如果电压上升的斜率小于 0.5 V/ms（最小值），则输入一个低电平到电源的 $\overline{\text{RESET}}$ 引脚，直至电压达到 1.8V，或者通过使用选项字节（POCMODE = 1）设置 2.7V/1.59V POC 模式（见图 5-14）。通过这种方式，在由 $\overline{\text{RESET}}$ 引脚释放复位后，CPU 按照图 5-13 中的<2> 及之后的时序进行操作。

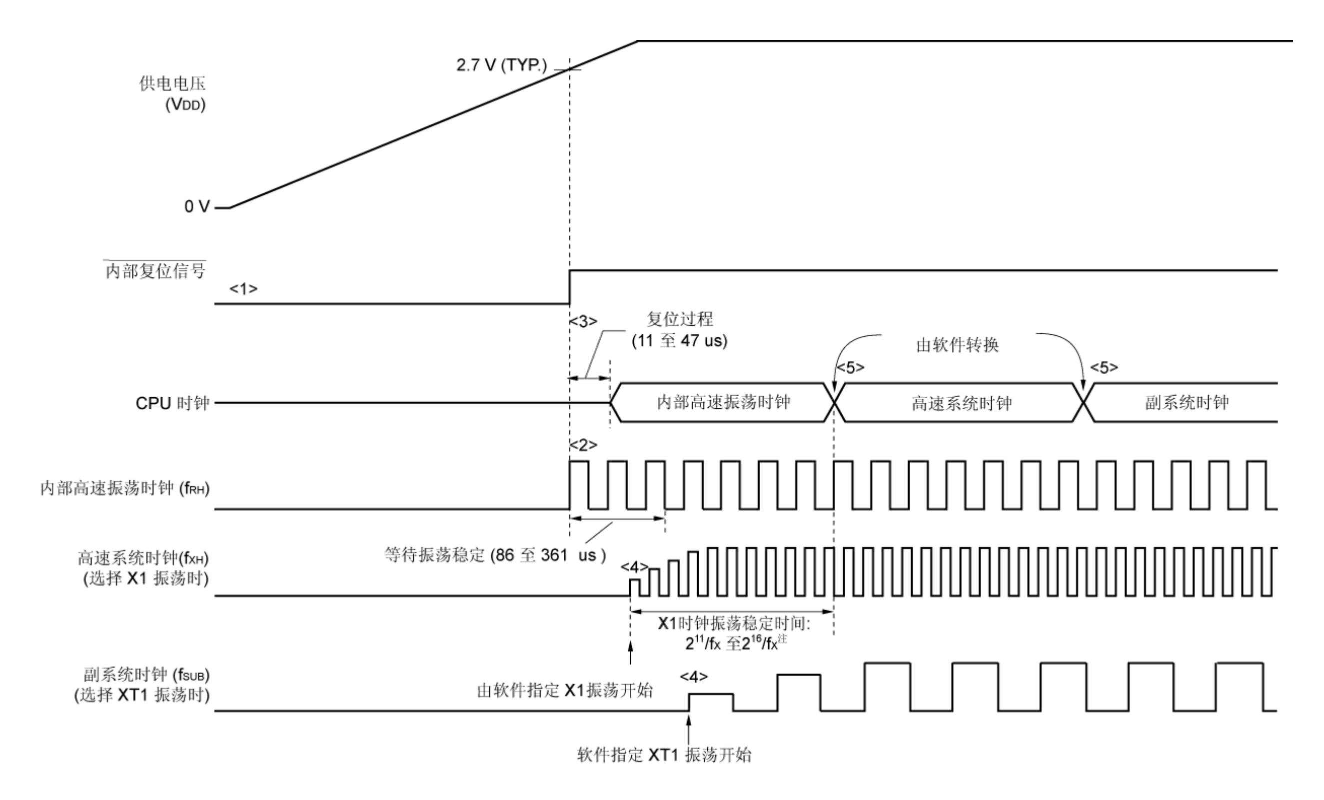
2. 当使用从 EXCLK 引脚输入的外部时钟时，不需要等待振荡稳定时间。

备注　　当微控制器正在操作时，不用作 CPU 时钟的时钟可以通过软件设置来停止。内部高速振荡时钟和高速系统时钟可以通过执行 STOP 指令来停止（参见 **5.6.1 高速系统时钟控制示例**中的（**4**）和 **5.6.3 副系统时钟控制示例**中的（**4**））。

图 **5-14.** 当打开供电电源上电时的时钟发生器操作



（设置为 **2.7 V/1.59 V POC** 模式 （选项字节： **POCMODE = 1**））



<1>　当电源打开时，通过上电清零 （POC） 电路产生一个内部复位信号。

<2>　当供电电压超过 2.7 V（典型值）时，复位被释放，且内部高速振荡器自动开始振荡。

<3>　复位释放并执行复位处理后，CPU 开始运行于内部高速振荡时钟操作。

<4>　通过软件设置 X1 或 XT1 时钟开始振荡（参见 **5.6.1 高速系统时钟控制示例**中的（**1**）和 **5.6.3 副系统时钟控制示例**中的（**1**））。

<5>　当 CPU 时钟切换到 X1 或 XT1 时钟时，等待时钟振荡稳定，然后通过软件设置切换（参见 **5.6.1 高速系统时钟控制示例**中的（**3**）和 **5.6.3 副系统时钟控制示例**中的（**3**））。

注　　复位释放时（如上图所示）或在 CPU 运行于内部高速振荡时钟时释放 STOP 模式，使用振荡稳定时间计数器状态寄存器（OSTC）来确认 X1 时钟的振荡稳定时间。如果 CPU 运行于高速系统时钟（X1 振荡），则可以使用振荡稳定时间选择寄存器（OSTS）来设置释放 STOP 模式时的振荡稳定时间。

<R>　注意事项　**1.** **在供电电压达到 1.59V（典型值）后，必须有 1.93 至 5.39 ms 的电压振荡稳定时间。如果在 1.93 至 5.39 ms 之内电源电压从 1.59 V（典型值）上升到 2.7 V（典型值），则在复位处理前自动产生 0 至 5.39 ms 的供电电源稳定等待时间，复位处理时间变为 19 至 80 μs。**

2. **当使用从 EXCLK 引脚输入的外部时钟时，不需要等待振荡稳定时间。**

备注　当微控制器正在操作时，不用作 CPU 时钟的时钟可以通过软件设置来停止。内部高速振荡时钟和高速系统时钟可以通过执行 STOP 指令来停止（参见 **5.6.1 高速系统时钟控制示例**中的（**4**）和 **5.6.3 副系统时钟控制示例**中的（**4**））。

5.6 时钟控制

5.6.1 控制高速系统时钟示例

有如下两种高速系统时钟可供使用。

- X1 时钟：　晶体/陶瓷谐振器连接到 X1 和 X2 引脚。
- 外部主系统时钟：　输入到 EXCLK 引脚的外部时钟。

不使用高速系统时钟时，OCD0A/X1/P121 和 OCD0B/X2/EXCLK/P122 引脚可以用作 I/O 端口引脚。

注意事项　复位释放后，OCD0A/X1/P121 和 OCD0B/X2/EXCLK/P122 引脚被设置为 I/O 端口模式。

下面描述的是下列情况时的设置过程示例。

（1）当 X1 时钟振荡时
（2）当使用外部主系统时钟时
（3）当高速系统时钟被作为 CPU 时钟和外设硬件时钟时
（4）当高速系统时钟停止时

（1）当 X1 时钟振荡时的设置过程示例

<1>　设置 P121/X1 和 P122/X2/EXCLK 引脚，并选择 X1 时钟或外部时钟（OSCCTL 寄存器），当 EXCLK 被清除为 0 且 OSCSEL 被设置为 1 时，从端口模式切换到 X1 振荡模式

EXCLK	OSCSEL	高速系统时钟引脚操作模式	P121/X1 引脚	P122/X2/EXCLK 引脚
0	1	X1 振荡模式	晶体/陶瓷振荡器连接	

<2>　控制 X1 时钟的振荡 （MOC 寄存器）
若 MSTOP 被清除为 0，则 X1 振荡器开始振荡。

<3>　等待 X1 时钟振荡的稳定
检查 OSTC 寄存器并等待所需的时间。
在等待时间内，可以使用内部高速振荡时钟执行其它软件处理。

注意事项　1.　在 X1 时钟运行时，不要改变 EXCLK 和 OSCSEL 的值。
2.　当供电电压达到所用时钟的操作电压后，设置 X1 时钟（参见 第三十章　电气特性（标准产品））。

（2）当使用外部主系统时钟时的设置过程示例

<1> 设置 P121/X1 和 P122/X2/EXCLK 引脚，并选择操作模式（OSCCTL 寄存器）

当 EXCLK 和 OSCSEL 被设置为 1 时，从端口模式切换到外部时钟输入模式。

EXCLK	OSCSEL	高速系统时钟引脚操作模式	P121/X1 引脚	P122/X2/EXCLK 引脚
1	1	外部时钟输入模式	I/O 端口	外部时钟输入

<2> 控制外部主系统时钟输入（MOC 寄存器）

MSTOP 被清除为 0 时，允许外部主系统时钟的输入。

注意事项 1. 在 X1 时钟运行时，不要改变 EXCLK 和 OSCSEL 的值。

2. 当供电电压达到所用时钟的操作电压后，设置 X1 时钟（参见 第三十章 电气特性（标准产品））。

（3）当高速系统时钟被用作 CPU 时钟和外设硬件时钟时的设置过程示例

<1> 设置高速系统时钟振荡[注]

（参见 **5.6.1 （1） 当 X1 时钟振荡时的设置过程示例**和 **（2） 当使用外部主系统时钟时的设置过程示例。**）

注 当高速系统时钟已经运行时，不需要设置<1>。

<2> 设置高速系统时钟作为主系统时钟（MCM 寄存器）

当 XSEL 和 MCM0 被设置为 1 时，高速系统时钟作为主系统时钟和外设硬件时钟 。

XSEL	MCM0	主系统时钟的选择和提供给外设硬件的时钟	
		主系统时钟 （f_{XP}）	外设硬件时钟 （f_{PRS}）
1	1	高速系统时钟 （f_{XH}）	高速系统时钟 （f_{XH}）

注意事项 如果高速系统时钟被选择用作主系统时钟，则除高速系统时钟之外的时钟不能被设置作为外设硬件时钟。

<3> 设置主系统时钟作为 CPU 时钟并选择分频比（PCC 寄存器）

当 CSS 被清除为 0 时，主系统时钟提供给 CPU。若要选择 CPU 时钟分频比，请使用 PCC0、PCC1 和 PCC2。

CSS	PCC2	PCC1	PCC0	CPU 时钟 （f_{CPU}） 选择
0	0	0	0	f_{XP}
	0	0	1	$f_{XP}/2$ （默认）
	0	1	0	$f_{XP}/2^2$
	0	1	1	$f_{XP}/2^3$
	1	0	0	$f_{XP}/2^4$
	其它情况			禁止设置

（4）当高速系统时钟停止时的设置过程示例

高速系统时钟可以通过如下两种方式停止。

- 执行 STOP 指令设置为 STOP 模式
- MSTOP 设置为 1，并停止 X1 振荡（如果使用外部时钟，则禁止时钟输入）

（a） 执行 STOP 指令

<1> 设置停止外设硬件

停止那些不能在 STOP 模式下使用的外设硬件（不能在 STOP 模式下使用的外设硬件，参见**第二十二章 待机功能** ）

<2> 设置待机模式释放后的 X1 时钟振荡稳定时间

当 CPU 运行于 X1 时钟时，在执行 STOP 指令之前，设置 OSTS 寄存器的值。

<3> 执行 STOP 指令

执行 STOP 指令后，系统处于 STOP 模式下，且 X1 振荡停止 （禁止外部时钟的输入）。

（b） 通过将 MSTOP 设置为 1 来停止 X1 振荡（禁止外部时钟输入）

<1> 确认 CPU 时钟状态（PCC 和 MCM 寄存器）

根据 CLS 和 MCS 确认 CPU 运行于高速系统时钟之外的时钟。

当 CLS = 0 且 MCS = 1 时，高速系统时钟提供给 CPU，因此将 CPU 时钟切换到副系统时钟或者内部高速振荡时钟。

CLS	MCS	CPU 时钟状态
0	0	内部高速振荡时钟
0	1	高速系统时钟
1	×	副系统时钟

<2> 停止高速系统时钟（MOC 寄存器）

当 MSTOP 被设置为 1 时，X1 振荡停止（禁止外部时钟的输入）。

注意事项　当设置 MSTOP = 1 时，请务必确认 MCS = 0 或 CLS = 1。此外，必须停止正在使用高速系统时钟的外设硬件。

5.6.2 控制内部高速振荡时钟示例

下面描述的是下列情况时的设置过程示例。

（1） 当内部高速振荡时钟重新开始振荡时

（2） 当内部高速振荡时钟被用作 CPU 时钟，并且内部高速振荡时钟或高速系统时钟被用作外设硬件时钟时

（3） 当内部高速振荡时钟停止时

（1）当内部高速振荡时钟重新振荡时的设置过程示例[注1]

<1> 内部高速振荡时钟重新振荡的设置（RCM 寄存器）

当 RSTOP 被清除为 0 时，内部高速振荡时钟开始运行。

<2> 等待内部高速振荡时钟的振荡精确稳定时间（RCM 寄存器）。

一直等待，直到 RSTS 被设置为 1[注2]。

注　**1.** 复位释放后，内部高速振荡器自动开始振荡，且内部高速振荡时钟被选择作为 CPU 时钟。

2. 如果 CPU 时钟和外设硬件时钟不需要很高的精度，则无需等待时间。

（2）当使用内部高速振荡时钟作为 CPU 时钟且内部高速振荡时钟或高速系统时钟作为外设硬件时钟时的设置过程示例

<1>
- 内部高速振荡时钟重新振荡[注]
  （参见 **5.6.2（1）当内部高速振荡时钟重新振荡时的设置过程示例**）。
- 高速系统时钟振荡[注]
  （使用高速系统时钟作为外设硬件时钟时所需的设置。参见 **5.6.1（1）当 X1 时钟振荡时的设置过程示例**和 **（2）当使用外部主系统时钟时的设置过程示例。**）

注　当内部高速振荡时钟或高速系统时钟已经运行时，不需要<1>的设置。

<2> 选择主系统时钟和外设硬件时钟（MCM 寄存器）

使用 XSEL 和 MCM0 设置主系统时钟和外设硬件时钟。

XSEL	MCM0	主系统时钟和外设硬件时钟的选择	
		主系统时钟（f_{XP}）	外设硬件时钟（f_{PRS}）
0	0	内部高速振荡时钟（f_{RH}）	内部高速振荡时钟（f_{RH}）
0	1		
1	0		高速系统时钟（f_{XH}）

<3> 选择 CPU 时钟分频比（PCC 寄存器）

当 CSS 被清除为 0 时，主系统时钟提供给 CPU。若要选择 CPU 时钟分频比，应该使用 PCC0、PCC1 和 PCC2。

CSS	PCC2	PCC1	PCC0	CPU 时钟（f_{CPU}）选择
0	0	0	0	f_{XP}
	0	0	1	$f_{XP}/2$（默认）
	0	1	0	$f_{XP}/2^2$
	0	1	1	$f_{XP}/2^3$
	1	0	0	$f_{XP}/2^4$
	其它情况			禁止设置

（3）当内部高速振荡时钟停止时的设置过程示例

内部高速振荡时钟可以通过如下两种方式停止。

- 执行 STOP 指令，设置 STOP 模式
- 将 RSTOP 设置为 1，且停止内部高速振荡时钟

（a） 执行 STOP 指令

<1> 设置停止外设硬件

停止那些不能在 STOP 模式下使用的外设硬件（不能在 STOP 模式下使用的外设硬件，参见**第二十二章 待机功能**）

<2> 设置待机模式释放后的 X1 时钟振荡稳定时间

当 CPU 运行于 X1 时钟时，在执行 STOP 指令之前，设置 OSTS 寄存器的值。

<3> 执行 STOP 指令

执行 STOP 指令后，系统处于 STOP 模式下，且内部高速振荡时钟被停止。

（b） 通过将 RSTOP 设置为 1 来停止内部高速振荡时钟

<1> 确认 CPU 时钟状态（PCC 和 MCM 寄存器）

根据 CLS 和 MCS 确认 CPU 运行于内部高速振荡时钟之外的时钟。

当 CLS = 0 且 MCS = 0 时，内部高速振荡时钟提供给 CPU，因此将 CPU 时钟切换到高速系统时钟或副系统时钟

CLS	MCS	CPU 时钟状态
0	0	内部高速振荡时钟
0	1	高速系统时钟
1	×	副系统时钟

<2> 停止内部高速振荡时钟（RCM 寄存器）

当 RSTOP 被设置为 1 时，停止内部高速振荡时钟。

注意事项 当设置 RSTOP= 1 时，必须确认 MCS = 1 或 CLS = 1。此外，必须停止正在使用内部高速振荡时钟的外设硬件。

5.6.3 控制副系统时钟的示例

有如下两种类型的副系统时钟可供使用。

- XT1 时钟： 连接到 XT1 和 XT2 引脚的晶体/陶瓷振荡器。

不使用副系统时钟时， XT1/P123 和 XT2/P124 引脚可用作输入端口引脚。

注意事项 复位释放后，XT1/P123 和 XT2 /P124 引脚被设置为输入端口模式。

下面描述的是下列情况时的设置过程示例。

（1） 当 XT1 时钟振荡时

（2） 当副系统时钟被用作 CPU 时钟时

（3） 当副系统时钟停止时

（1） 当 XT1 时钟振荡时的设置过程示例

<1> 设置 XT1 和 XT2 引脚并选择操作模式 （PCC 和 OSCCTL 寄存器）

当 OSCSELS 被设置为下列任一模式时，将从端口模式切换到 XT1 振荡模式 。

OSCSELS	副系统时钟引脚操作模式	P123/XT1 引脚	P124/XT2 引脚
1	XT1 振荡模式	连接晶体/陶瓷谐振器	

<2> 等待副系统时钟振荡的稳定

通过软件等待副系统时钟振荡稳定时间，使用定时器功能。

注意事项　在副系统时钟运行时，不要改变 OSCSELS 的值。

（2） 当使用副系统时钟作为 CPU 时钟时的设置过程示例

<1> 设置副系统时钟振荡[注]

（参见 **5.6.3 （1） 当 XT1 时钟振荡时的设置过程示例**）

注　　当副系统时钟运行时，不需要<1>的设置。

<2> 切换 CPU 时钟（PCC 寄存器）

当 CSS 被设置为 1 时，副系统时钟提供给 CPU。

CSS	PCC2	PCC1	PCC0	CPU 时钟 （f_{CPU}） 选择
1	0	0	0	$f_{SUB}/2$
	0	0	1	
	0	1	0	
	0	1	1	
	1	0	0	
	其它情况			禁止设置

（3）当副系统时钟停止时的设置过程示例

<1> 确认 CPU 时钟状态（PCC 和 MCM 寄存器）

根据 CLS 和 MCS 确认 CPU 运行于副系统时钟 之外的时钟。

当 CLS = 1 时，副系统时钟提供给 CPU，因此将 CPU 时钟切换到内部高速振荡时钟或高速系统时钟。

CLS	MCS	CPU 时钟状态
0	0	内部高速振荡时钟
0	1	高速系统时钟
1	×	副系统时钟

<2> 停止副系统时钟 （OSCCTL 寄存器）

当 OSCSELS 被清除为 0 时，停止 XT1 振荡。

注意事项　**1. 当清除 OSCSEL 为 0 时，必须确认 CLS = 0。此外，应该停止正在使用副系统时钟的外设硬件。**

2. 使用 STOP 指令，不能停止副系统时钟的振荡。

5.6.4 控制内部低速振荡时钟的示例

内部低速振荡时钟不能被用作 CPU 时钟。

只有下列外设硬件可以使用该时钟。

- 看门狗定时器
- 8 位定时器 H1（如果 f_{RL}，$f_{RL}/2^7$ 或 $f_{RL}/2^9$ 被选择作为计数时钟）
- LCD 控制器/驱动器（如果 $f_{RL}/2^3$ 被选择作为 LCD 源时钟）

此外，可以通过选项字节选择如下操作模式。

- 内部低速振荡器不能停止
- 内部低速振荡器可由软件停止

复位释放后，内部低速振荡器自动开始振荡，如果使用选项字节允许看门狗定时器操作，则可以驱动看门狗定时器（240 kHz （典型值））。

（1） 当内部低速振荡时钟停止时的设置过程示例

<1> 将 LSRSTOP 置 1（RCM 寄存器）

当 LSRSTOP 被设置为 1 时，内部低速振荡时钟被停止。

（2） 内部低速振荡时钟重新振荡时设置过程示例

<1> 将 LSRSTOP 清零（RCM 寄存器）

当 LSRSTOP 被清除为 0 时，内部低速振荡时钟重新振荡。

注意事项　如果通过选项字节选择为"不能停止内部低速振荡器"，则不能控制内部低速振荡时钟的振荡。

5.6.5 提供给CPU和外设硬件的时钟

下表显示了提供给 CPU 和外设硬件的时钟之间的关系及寄存器的设置。

表 5-4.　提供给 CPU 和外设硬件的时钟及寄存器设置

提供的时钟		XSEL	CSS	MCM0	EXCLK
提供给 CPU 的时钟	提供给外设硬件的时钟				
内部高速振荡时钟		0	0	×	×
内部高速振荡时钟	X1 时钟	1	0	0	0
	外部主系统时钟	1	0	0	1
X1 时钟		1	0	1	0
外部主系统时钟		1	0	1	1
副系统时钟	内部高速振荡时钟	0	1	×	×
	X1 时钟	1	1	0	0
		1	1	1	0
	外部主系统时钟	1	1	0	1
		1	1	1	1

备注　**1.** XSEL：　主时钟模式寄存器（MCM）的第 2 位

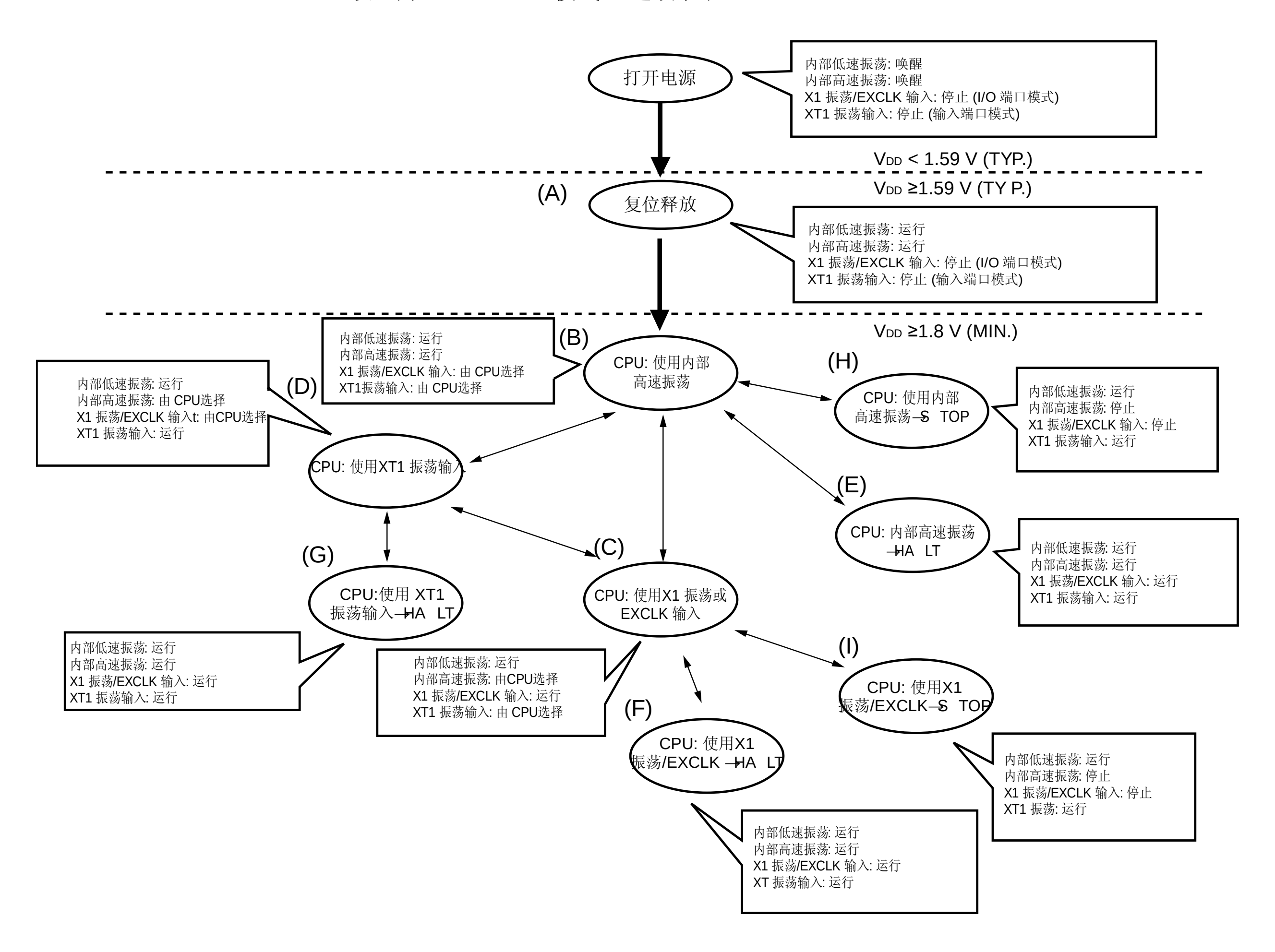
2. CSS：　处理器时钟控制寄存器（PCC）的第 4 位
3. MCM0：　MCM 的第 0 位
4. EXCLK：　时钟操作模式选择寄存器（OSCCTL）的第 7 位
5. ×：　无须理会

5.6.6 CPU 时钟状态迁移图

图 5-15 显示了该产品 CPU 时钟状态迁移图。

图 5-15. CPU 时钟状态迁移图

（设置为 1.59 V POC 模式 （选项字节：POCMODE = 0））



备注 在 2.7V/1.59V POC 模式下（选项字节：POCMODE = 1），当供电电压超过 2.7V（典型值）时，CPU 时钟状态改变为上图中的（A），而在复位处理后（11 至 47 μs （典型值））改变为（B）。

表 5-5 显示了 CPU 时钟的切换与 SFR 寄存器的设置示例

表 5-5. CPU 时钟切换与 SFR 寄存器设置示例（1/4）

（1）　复位释放后（A）CPU 运行于内部高速振荡时钟（B）

状态迁移	SFR 寄存器设置
（A）→（B）	SFR 寄存器不必设置 （复位释放后默认状态）.

（2）　复位释放后（A）CPU 运行于高速系统时钟（C）
（复位释放后，CPU 立即使用内部高速振荡时钟（B））

（SFR 寄存器设置顺序）→

SFR 寄存器设置标志 / 状态迁移	EXCLK	OSCSEL	MSTOP	OSTC 寄存器	XSEL	MCM0
（A）→（B）→（C）（X1 时钟）	0	1	0	必须检查	1	1
（A）→（B）→（C）（外部主时钟）	1	1	0	不必检查	1	1

注意事项　供电电压达到所用时钟的操作电压后，设置时钟（参见 第三十章 电气特性（标准产品））。

(3)　复位释放后（A）CPU 运行于副系统时钟（D）
（复位释放后，CPU 立即使用内部高速振荡时钟（B））

（SFR 寄存器设置顺序）→

SFR 寄存器设置标志 / 状态迁移	OSCSELS	等待振荡稳定	CSS
（A）→（B）→（D）	1	必需	1

备注　**1.** 表 5-5 的（A）到（I）对应图 5-15 的（A）到（I）。
2. EXCLK，OSCSEL，OSCSELS：
时钟操作模式选择寄存器（OSCCTL）的第 7、6 位和第 4 位
MSTOP：主 OSC 控制寄存器（MOC）的第 7 位
XSEL，MCM0：主时钟模式寄存器（MCM）的第 2 位和第 0 位
CSS：处理器时钟控制寄存器（PCC）的第 4 位

表 5-5. CPU 时钟切换与 SFR 寄存器设置示例 （2/4）

（4） CPU 时钟从内部高速振荡时钟（B）切换到高速系统时钟（C）

（SFR 寄存器的设置顺序） →

状态迁移 \ SFR 寄存器的设置标志	EXCLK	OSCSEL	MSTOP	OSTC 寄存器	XSEL 注	MCM0
（B）→（C）（X1 时钟）	0	1	0	必须检查	1	1
（B）→（C）（外部主时钟）	1	1	0	不必检查	1	1

EXCLK、OSCSEL：如果这些寄存器都已经设置，则不必要
MSTOP、OSTC 寄存器：如果 CPU 运行于高速系统时钟，则不必要

注　复位释放后，该标志只能修改一次。如果已经设置了该项，则无需设置。

注意事项　供电电压达到所用时钟的操作电压后，设置时钟（参见 第三十章 电气特性（标准产品））。

（5） CPU 时钟从内部高速振荡时钟（B）切换到副系统时钟（D）

（SFR 寄存器的设置顺序） →

状态迁移 \ SFR 寄存器的设置标志	OSCSELS	等待振荡稳定	CSS
（B）→（D）	1	必须	1

备注　**1.** 表 5-5 的（A）到（I）对应图 5-15 的（A）到（I）。

2. EXCLK，OSCSEL，OSCSELS：
时钟操作模式选择寄存器（OSCCTL）的第 7、6 位和第 4 位
MSTOP：主 OSC 控制寄存器（MOC）的第 7 位
XSEL，MCM0：主时钟模式寄存器（MCM）的第 2 位和第 0 位
CSS：处理器时钟控制寄存器（PCC）的第 4 位

表 5-5. CPU 时钟切换与 SFR 寄存器设置示例 （3/4）

（6） CPU 时钟从高速系统时钟（C）切换到内部高速振荡时钟（B）

（SFR 寄存器的设置顺序） →

SFR 寄存器的设置标志 / 状态迁移	RSTOP	RSTS	MCM0
（C）→（B）	0	确认该标志为 1	0

如果 CPU 运行于内部高速振荡时钟，则不必要（RSTOP、RSTS）

（7）　CPU 时钟从高速系统时钟（C）切换到副系统时钟（D）

（ SFR 寄存器的设置顺序） →

SFR 寄存器的设置标志 / 状态迁移	OSCSELS	等待振荡稳定	CSS
（C）→（D）	1	必须	1

如果 CPU 运行于副系统时钟，则不必要（OSCSELS、等待振荡稳定）

（8） CPU 时钟从副系统时钟（D）切换到内部高速振荡时钟（B）

（SFR 寄存器的设置顺序） →

SFR 寄存器的设置标志 / 状态迁移	RSTOP	RSTS	MCM0	CSS
（D）→（B）	0	确认该标志为 1	0	0

如果 CPU 运行于内部高速振荡时钟，则不必要（RSTOP、RSTS）

如果 XSEL 为 0，则不必要（MCM0）

备注　**1.** 表 5-5 的（A）到（I）对应图 5-15 的（A）到（I）。

2. MCM0：　主时钟模式寄存器（MCM）的第 0 位
OSCSELS：　时钟操作模式选择寄存器（OSCCTL）的第 4 位
RSTS，RSTOP：　内部振荡模式寄存器（RCM）的第 7 位和第 0 位
CSS：　处理器时钟控制寄存器（PCC）的第 4 位

表 5-5. CPU 时钟切换与 SFR 寄存器设置示例 （4/4）

（9）　CPU 时钟从副系统时钟（D）切换到高速系统时钟（C）

（ SFR 寄存器的设置顺序） →

SFR 寄存器的设置标志 / 状态迁移	EXCLK	OSCSEL	MSTOP	OSTC 寄存器	XSEL 注	MCM0	CSS
（D）→（C）（X1 时钟）	0	1	0	必须检查	1	1	0
（D）→（C）（外部主时钟）	1	1	0	不必检查	1	1	0

EXCLK、OSCSEL：如果这些寄存器都已经设置，则不必要
MSTOP、OSTC 寄存器：如果 CPU 运行于高速系统时钟，则不必要

注　复位释放后，该标志只能修改一次。如果已经设置了该项，则无需再设置。

注意事项　供电电压达到所用时钟的操作电压后，设置时钟（参见 第三十章　电器特性（标准产品））。

（10）• CPU 运行于内部高速振荡时钟（B）时，设置 HALT 模式（E）。
• CPU 运行于高速系统时钟（C）时，设置 HALT 模式（F）。
• CPU 运行于副系统时钟（D）时，设置 HALT 模式（G）。

状态迁移	设置
（B）→（E） （C）→（F） （D）→（G）	执行 HALT 指令

（11）• CPU 运行于内部高速振荡时钟（B）时，设置 STOP 模式（H）。
• CPU 运行于高速系统时钟（C）时，设置 STOP 模式（I）。

（设置顺序） →

状态迁移	设置	
（B）→（H） （C）→（I）	停止那些不能在 STOP 模式下操作的外设功能	执行 STOP 指令

备注　**1.** 表 5-5 的（A）到（I）对应图 5-15 的（A）到（I）。
2. EXCLK，OSCSEL：　时钟操作模式选择寄存器（OSCCTL）的第 7 和 6 位
MSTOP：　主 OSC 控制寄存器（MOC）的第 7 位
XSEL，MCM0：　主时钟模式寄存器（MCM）的第 2 位和第 0 位
CSS：　处理器时钟控制寄存器（PCC） 的第 4 位

5.6.7 CPU时钟切换之前的条件与CPU时钟改变之后的处理

CPU 时钟切换之前的条件与 CPU 时钟改变之后的处理展示如下。

表 5-6. 改变 CPU 时钟

<table>
<tr><th colspan="2">CPU 时钟</th><th rowspan="2">改变前的条件</th><th rowspan="2">改变后的处理</th></tr>
<tr><th>改变前</th><th>改变后</th></tr>
<tr><td rowspan="2">内部高速振荡时钟</td><td>X1 时钟</td><td>X1 振荡的稳定
• MSTOP = 0，OSCSEL = 1，EXCLK = 0
• 经过振荡稳定时间之后</td><td rowspan="2">• 内部高速振荡器可以停止（RSTOP = 1）.</td></tr>
<tr><td>外部主系统时钟</td><td>允许来自 EXCLK 引脚的外部时钟的输入
• MSTOP = 0，OSCSEL = 1，EXCLK = 1</td></tr>
<tr><td>X1 时钟</td><td rowspan="2">内部高速振荡时钟</td><td rowspan="2">内部高速振荡器的振荡
• RSTOP = 0</td><td>X1 振荡可以停止（MSTOP = 1）.</td></tr>
<tr><td>外部主系统时钟</td><td>外部主系统时钟的输入可以禁止（MSTOP = 1）</td></tr>
<tr><td>内部高速振荡时钟</td><td rowspan="3">XT1 时钟</td><td rowspan="3">XT1 振荡的稳定
• OSCSELS = 1
• 经过振荡稳定时间之后</td><td>停止内部高速振荡器（RSTOP = 1）可以降低操作电流</td></tr>
<tr><td>X1 时钟</td><td>X1 振荡可以停止（MSTOP = 1）</td></tr>
<tr><td>外部主系统时钟</td><td>外部主系统时钟的输入可以禁止（MSTOP = 1）</td></tr>
<tr><td rowspan="3">XT1 时钟</td><td>内部高速振荡时钟</td><td>内部高速振荡器振荡，且选择内部高速振荡时钟作为主系统时钟
• RSTOP = 0，MCS = 0</td><td rowspan="3">XT1 振荡可以停止（OSCSELS = 0）.</td></tr>
<tr><td>X1 时钟</td><td>X1 振荡稳定，且选择高速系统时钟作为主系统时钟
• MSTOP = 0，OSCSEL = 1，EXCLK = 0
• 经过振荡稳定时间之后
• MCS = 1</td></tr>
<tr><td>外部主系统时钟</td><td>允许来自 EXCLK 引脚的外部时钟输入，并选择高速系统时钟作为主系统时钟
• MSTOP = 0，OSCSEL = 1，EXCLK = 1
• MCS = 1</td></tr>
</table>

5.6.8 CPU时钟和主系统时钟切换所需的时间

通过设置处理器时钟控制器（PCC）的第 0 位到第 2 位（PCC0 到 PCC2）以及第 4 位（CSS），可以切换 CPU 时钟（在主系统时钟和副系统时钟之间），也可以改变主系统时钟的分频比。

重写 PCC 后，实际的切换操作不会立即执行；使用切换前的时钟，继续操作几个时钟（参见 **表 5-7**）。

可以使用 PCC 寄存器的第 5 位（CLS）来确定 CPU 运行于主系统时钟还是副系统时钟。

表 5-7.　CPU 时钟切换所需时间与主系统时钟周期分频因数

<table>
<tr><th colspan="4">切换前的设置值</th><th colspan="24">切换后的设置值</th></tr>
<tr><td>CSS</td><td>PCC2</td><td>PCC1</td><td>PCC0</td><td>CSS</td><td>PCC2</td><td>PCC1</td><td>PCC0</td><td>CSS</td><td>PCC2</td><td>PCC1</td><td>PCC0</td><td>CSS</td><td>PCC2</td><td>PCC1</td><td>PCC0</td><td>CSS</td><td>PCC2</td><td>PCC1</td><td>PCC0</td><td>CSS</td><td>PCC2</td><td>PCC1</td><td>PCC0</td><td>CSS</td><td>PCC2</td><td>PCC1</td><td>PCC0</td></tr>
<tr><td></td><td></td><td></td><td></td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>×</td><td>×</td><td>×</td></tr>
<tr><td rowspan="5">0</td><td>0</td><td>0</td><td>0</td><td colspan="4"></td><td colspan="4">16 个时钟</td><td colspan="4">16 个时钟</td><td colspan="4">16 个时钟</td><td colspan="4">16 个时钟</td><td colspan="4">2fXP/fSUB 个时钟</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td colspan="4">8 个时钟</td><td colspan="4"></td><td colspan="4">8 个时钟</td><td colspan="4">8 个时钟</td><td colspan="4">8 个时钟</td><td colspan="4">fXP/fSUB 个时钟</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td colspan="4">4 个时钟</td><td colspan="4">4 个时钟</td><td colspan="4"></td><td colspan="4">4 个时钟</td><td colspan="4">4 个时钟</td><td colspan="4">fXP/2fSUB 个时钟</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td colspan="4">2 个时钟</td><td colspan="4">2 个时钟</td><td colspan="4">2 个时钟</td><td colspan="4"></td><td colspan="4">2 个时钟</td><td colspan="4">fXP/4fSUB 个时钟</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td colspan="4">1 个时钟</td><td colspan="4">1 个时钟</td><td colspan="4">1 个时钟</td><td colspan="4">1 个时钟</td><td colspan="4"></td><td colspan="4">fXP/8fSUB 个时钟</td></tr>
<tr><td>1</td><td>×</td><td>×</td><td>×</td><td colspan="4">2 个时钟</td><td colspan="4">2 个时钟</td><td colspan="4">2 个时钟</td><td colspan="4">2 个时钟</td><td colspan="4">2 个时钟</td><td colspan="4"></td></tr>
</table>

注意事项　**主系统时钟周期分频因数（PCC0 到 PCC2）的选择，和主系统时钟到副系统时钟的切换（CSS 从 0 改变为 1）不能同时设置。**

但对于主系统时钟周期分频因数（PCC0 到 PCC2）的选择，和副系统时钟到主系统时钟的切换（CSS 从 1 改变为 0）可以同时设置。

备注　**1.** 表 5-7 列出的时钟数量是切换前的 CPU 时钟数量。

2. 将 CPU 时钟从主系统时钟切换到副系统时钟，计算时钟数量时，可以向上舍入一个时钟并丢弃小数部分，如下所示。

示例：　当 CPU 时钟从 $f_{XP}/2$ 切换到 $f_{SUB}/2$ 时（@ 振荡频率 f_{SUB} = 32.768 kHz，f_{XP} = 10 MHz）

$f_{XP}/f_{SUB} = 10000/32.768 \cong 305.1 \rightarrow 306$ 个时钟

通过设置主时钟模式寄存器（MCM）的第 0 位（MCM0），可以切换主系统时钟（在内部高速振荡时钟和高速系统时钟之间）。

重写 MCM0 后，实际的切换操作不会立即执行；使用切换前的时钟，继续操作几个时钟（见 **表 5-8**）。

可以使用 MCM 寄存器的第 1 位（MCS）来确定 CPU 运行于内部高速振荡时钟还是高速系统时钟。

表 5-8.　主系统时钟切换所需的最长时间

切换前的设置值	切换后的设置值	
MCM0	MCM0	
	0	1
0		$1 + 2f_{RH}/f_{XH}$ 时钟
1	$1 + 2f_{XH}/f_{RH}$ 时钟	

注意事项　**当内部高速振荡时钟切换到高速系统时钟时，MCM 的第 2 位（XSEL）必须预先设置为 1。XSEL 的值在复位释放后只能修改一次。**

备注　**1.** 表 5-8 所列时钟数量是切换之前的主系统时钟数量。

2. 计算表 5-8 的时钟数时，丢弃小数部分。

示例：　当主系统时钟从内部高速振荡时钟切换到高速系统时钟时（@振荡频率 f_{RH} = 8 MHz，f_{XH} = 10 MHz）

$$1 + 2f_{RH}/f_{XH} = 1 + 2 \times 8/10 = 1 + 2 \times 0.8 = 1 + 1.6 = 2.6 \rightarrow 2 \text{ 个时钟}$$

5.6.9 时钟振荡停止前的条件

下表列出了停止时钟振荡（禁止外部时钟输入）的寄存器标志位设置和时钟振荡停止前的条件。

表 5-9.　时钟振荡停止前的条件和标志位设置

时钟	时钟振荡停止前的条件 （禁止外部时钟输入）	SFR 寄存器的 标志位设置
内部高速振荡时钟	MCS = 1 或 CLS = 1 （CPU 运行于内部高速振荡时钟之外的时钟）	RSTOP = 1
X1 时钟 外部主系统时钟	MCS = 0 或 CLS = 1 （CPU 运行于高速系统时钟之外的时钟）	MSTOP = 1
XT1 时钟	CLS = 0 （CPU 运行于副系统时钟之外的时钟）	OSCSELS = 0

5.6.10 外设硬件与源时钟

下表列出了 78K0/LE3 中包含的外设硬件与源时钟。

表 5-10. 外设硬件与源时钟

外设硬件 \ 源时钟		外设硬件时钟（f_{PRS}）	副系统时钟（f_{SUB}）	内部低速振荡时钟（f_{RL}）	TM50 输出	TM52 输出	TMH1 输出	来自外设硬件引脚的外部时钟
16 位定时器/事件计数器	00	Y	Y	N	N	Y	N	Y (TI000 引脚)[注]
8 位定时器/事件计数器	50	Y	N	N	N	N	N	Y (TI50 引脚)[注]
	51	Y	N	N	N	N	Y	Y (TI51 引脚)[注]
	52	Y	N	N	N	N	N	Y (TI52 引脚)[注]
8 位定时器	H0	Y	N	N	Y	N	N	N
	H1	Y	N	Y	N	N	N	N
	H2	Y	N	N	N	N	N	N
实时计数器		Y	Y	N	N	N	N	N
看门狗定时器		N	N	Y	N	N	N	N
蜂鸣器输出		Y	N	N	N	N	N	N
逐次逼近型 A/D 转换器		Y	N	N	N	N	N	N
ΔΣ 型 A/D 转换器		Y	Y	N	N	N	N	N
串行接口	UART0	Y	N	N	Y	N	N	N
	UART6	Y	N	N	Y	N	N	N
	CSI10	Y	N	N	N	N	N	Y ($\overline{SCK10}$ 引脚)[注]
LCD 控制器/驱动器		Y	Y	Y	N	N	N	N
曼彻斯特编码发生器		Y	N	N	N	N	N	N
遥控接收器		Y	Y	N	N	N	N	N

注　当 CPU 运行于副系统时钟且内部高速振荡时钟已经停止时，使用由外设硬件引脚输入的外部时钟的这些功能，不要启动。

备注　Y：可以选择，N：不可选择

6.1 16 位定时器/事件计数器 00 的功能

16 位定时器/事件计数器 00 具有以下功能。

（1） 间隔定时器

16 位定时器/事件计数器 00 以预置的时间间隔产生中断请求。

（2） 方波输出

16 位定时器/事件计数器 00 可以输出任意选择频率的方波。

（3） 外部事件计数器

16 位定时器/事件计数器 00 可以测量外部输入信号的脉冲数量。

（4） 单次触发脉冲输出

16 位定时器/事件计数器 00 可以输出单次触发脉冲，其中脉冲宽度可以任意设置。

（5） PPG 输出

16 位定时器/事件计数器 00 可以输出矩形波，其中矩形波的频率和输出宽度可以自由设置。

（6） 脉冲宽度测量

16 位定时器/事件计数器 00 可以测量外部输入信号的脉冲宽度。

（7） 外部 24 位事件计数器

通过级联 16 位定时器 00 和 8 位定时器/事件计数器 52，并使用 8 位定时器/事件计数器 52 的外部事件计数器功能，16 位定时器/事件计数器 00 可作为外部 24 位事件计数器进行操作。

当用作外部 24 位事件计数器时，可以由 8 位定时计数器 H2 输出来控制外部事件输入门的使能。

6.2 16 位定时器/事件计数器 00 的配置

16 位定时器/事件计数器 00 包括以下硬件。

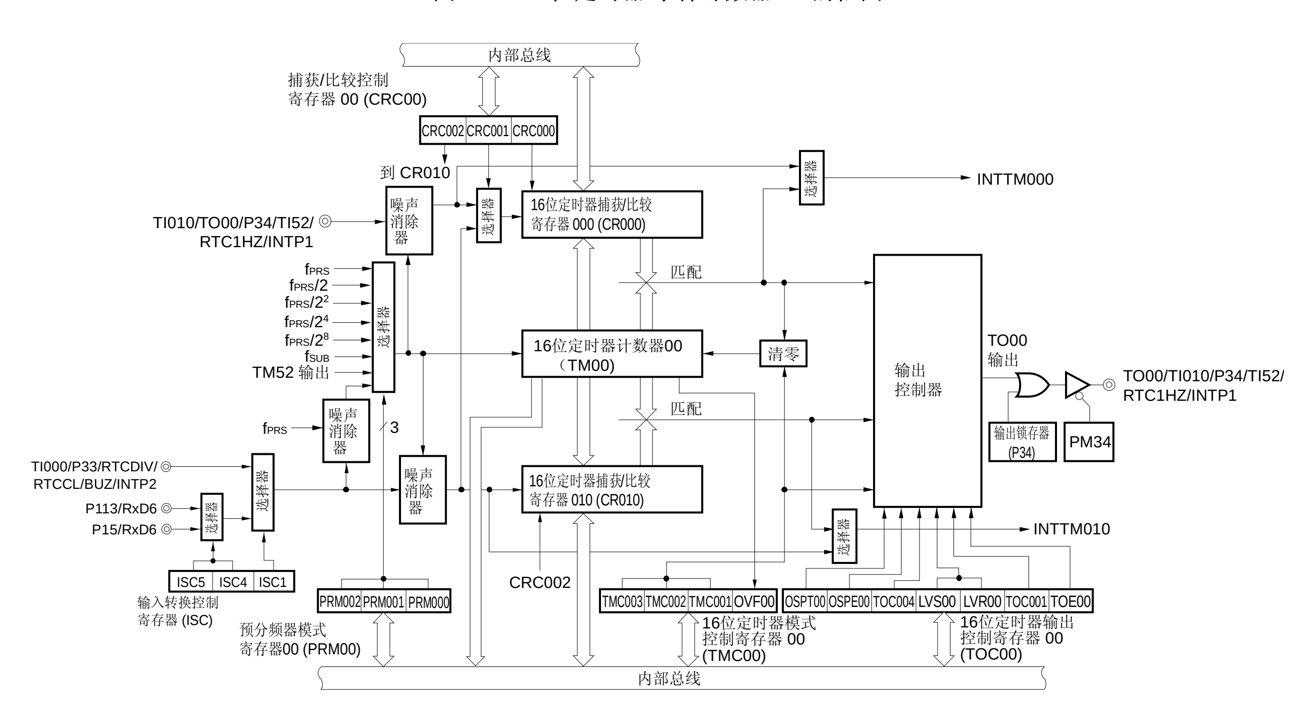
表 6-1. 16 位定时器/事件计数器 00 的配置

项目	配置
定时器/计数器	16 位定时器计数器 00（TM00）
寄存器	16 位定时器捕获/比较寄存器 000，010（CR000，CR010）
定时器输入	TI000，TI010 引脚
定时器输出	TO00 引脚，输出控制器
控制寄存器	16 位定时器模式控制寄存器 00（TMC00） 捕获/比较控制寄存器 00（CRC00） 16 位定时器输出控制寄存器 00（TOC00） 预分频器模式寄存器 00（PRM00） 输入切换控制寄存器（ISC） 端口模式寄存器 3（PM3） 端口寄存器 3（P3）

备注 当使用 16 位定时器/事件计数器 00 作为外部 24 位事件计数器时，要使用 8 位定时器/事件计数器 52（TM52）和 8 位定时计数器 H2（TMH2）。详情参见 **6.4.9 外部 24 位事件计数器操作。**

图 6-1 显示了 16 位定时器/事件计数器的框图。

图 6-1. 16 位定时器/事件计数器 00 的框图



注意事项 1. P34 引脚不能同时用作 TI010 有效边沿和定时器输出（TO00）。请选择其中一个功能。

注意事项 2. 如果将 16 位定时器模式控制寄存器 00（TMC00）的第 3 位和第 2 位（TMC003 和 TMC002）清除为 00 的时序和捕获触发输入发生冲突，则捕获到的数据不确定。

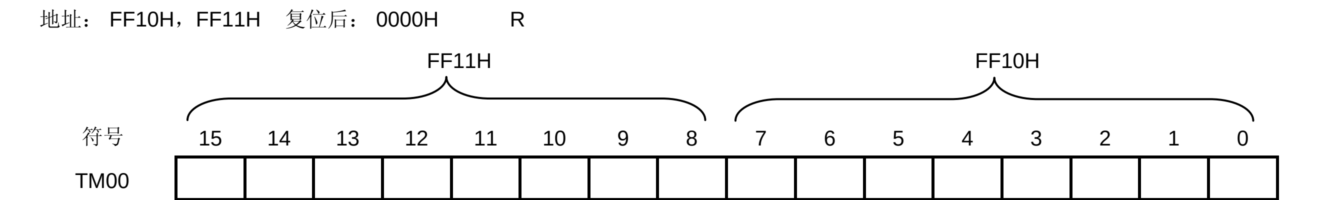
3. 要从捕获模式改变为比较模式，首先将 TMC003 和 TMC002 位清除为 00，然后再改变设置。曾经被捕获的值一直保存在 CR000，除非设备复位。如果将模式改变为比较模式，请确保设置一个比较值。

（1） 16 位定时器计数器 00（TM00）

TM00 是 16 位只读寄存器，用于对脉冲进行计数。
计数器随着计数时钟的上升沿同步增加。

图 **6-2. 16** 位定时器计数器 **00（TM00）**的格式

地址： FF10H，FF11H　复位后： 0000H　　R



当 16 位定时器模式控制寄存器 00（TMC00）的第 3 位和第 2 位不是 00 时，可以通过读取 TM00 来获取 TM00 的计数值。如果在 TMC003 和 TMC002 = 00 时读取 TM00，则获取的 TM00 计数值是 0000H。
在下列情况时，计数值复位为 0000H。

- 产生复位信号时
- 如果 TMC003 和 TMC002 被清除为 00
- 在有效边沿输入 TI000 引脚时发生清零&启动的模式下，如果有 TI000 引脚的有效边沿输入
- 在 TM00 和 CR000 匹配时发生清零&启动的模式下，如果 TM00 和 CR000 匹配
- 在单次触发脉冲输出模式下 OSPT00 被设置为 1，或有效边沿输入 TI000 引脚时

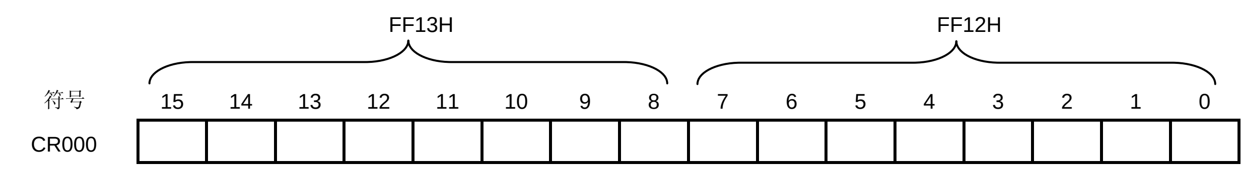
注意事项　即使 TM00 被读取，也不能通过 CR010 捕获到该值。

（2） 16 位定时器捕获/比较寄存器 000（CR000）

CR000 和 CR010 是 16 位寄存器，用于由 CRC00 选择的捕获或比较功能。
在定时器停止操作时（TMC003 和 TMC002 = 00），改变 CR000 的值。
如果已经按特定方式对 CR010 设置了该值，则在操作期间可以改变 CR010。详情参见 **6.5.1 TM00 操作期间重写 CR010**。
这些寄存器可以按照 16 位宽度进行读取或写入。
复位信号的产生会将这些寄存器设置为 0000H。

图 6-3. 16 位定时器捕获/比较寄存器 000（CR000）的格式

地址： FF12H，FF13H 复位后： 0000H R/W



（**i**） 当 **CR000** 用作比较寄存器

CR000 中的设置值总是与 TM00 计数值进行比较，如果两者匹配，则产生一个中断请求信号（INTTM000）。在 CR000 被重写之前，该值一直被保持。

注意事项 当 CR000 被设置为比较模式时，不能执行捕获操作，即使有捕获触发输入。

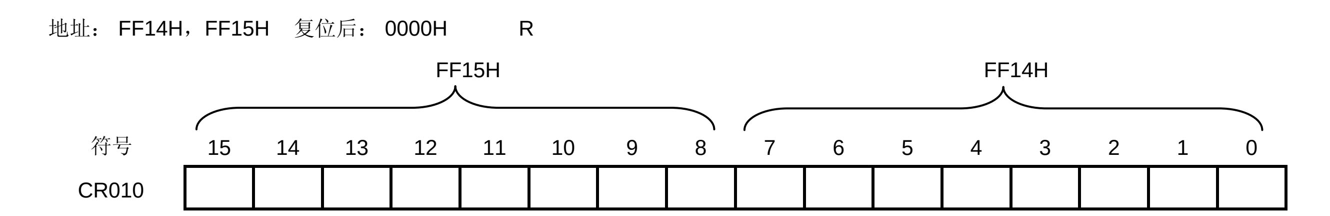
（**ii**） 当 **CR000** 用作捕获寄存器

当捕获触发输入时，TM00 的计数值被捕获到 CR000。

TI000 引脚电平反向的脉冲沿或 TI010 引脚的有效边沿，都可以通过 CRC00 或 PRM00 选择作为捕获触发信号。

图 6-4. 16 位定时器 捕获/比较寄存器 010（CR010）的格式

地址： FF14H，FF15H 复位后： 0000H R



（**i**） 当 **CR010** 用作比较寄存器

CR010 中的设置值总是与 TM00 计数值进行比较，如果两者匹配，则产生一个中断请求信号（INTTM010）。

注意事项 当 CR010 被设置为比较模式时，不能执行捕获操作，即使有捕获触发输入。

（**ii**） 当 **CR010** 用作捕获寄存器

当捕获触发输入时，TM00 的计数值被捕获到 CR010。

可以选择 TI000 引脚的有效边沿作为捕获触发。TI000 有效边沿由 PRM00 置。

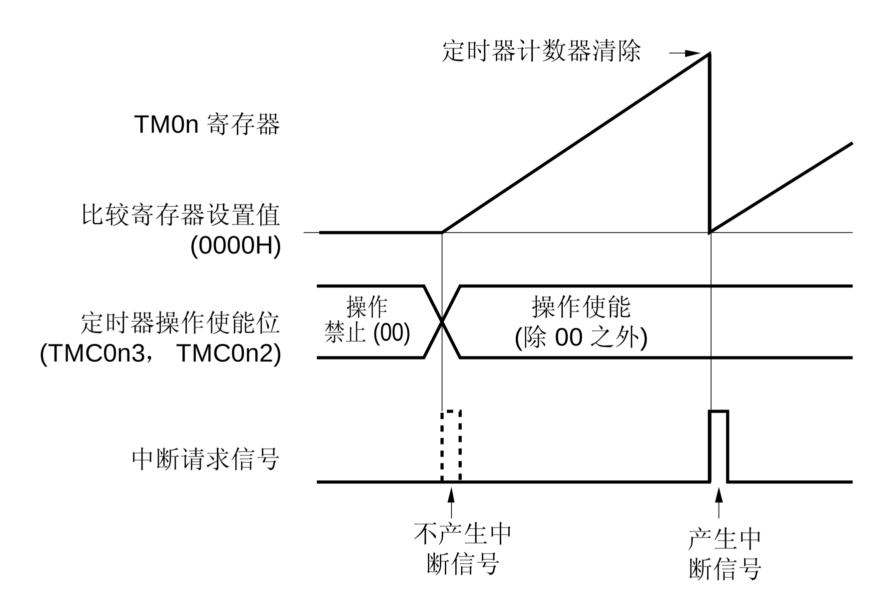
（iii） 当 CR000 或 CR010 用作比较寄存器时的设置范围

当 CR000 或 CR010 用作比较寄存器时，按下面所示进行设置。

操作	CR000 寄存器设置范围	CR010 寄存器设置范围
作为间隔定时器操作	0000H < N ≤ FFFFH	0000H[注] ≤ M ≤ FFFFH 通常不使用该设置。屏蔽匹配中断信号（INTTM010）。
作为方波输出操作		
作为外部事件计数器操作		
在有效边沿输入 TI000 引脚时进入清零&启动模式下操作	0000H[注] ≤ N ≤ FFFFH	0000H[注] ≤ M ≤ FFFFH
作为自由运行定时器操作		
作为 PPG 输出操作	M < N ≤ FFFFH	0000H[注] ≤ M < N
作为单次触发脉冲输出操作	0000H[注] ≤ N ≤ FFFFH（N ≠ M）	0000H[注] ≤ M ≤ FFFFH（M ≠ N）

注　当设置 0000H 时，定时器操作之后不会立即发生匹配中断，并且定时器输出不改变，第一个匹配时序如下所示。当定时计数器（TM00 寄存器）从 0000H 变为 0001H 时，产生匹配中断。

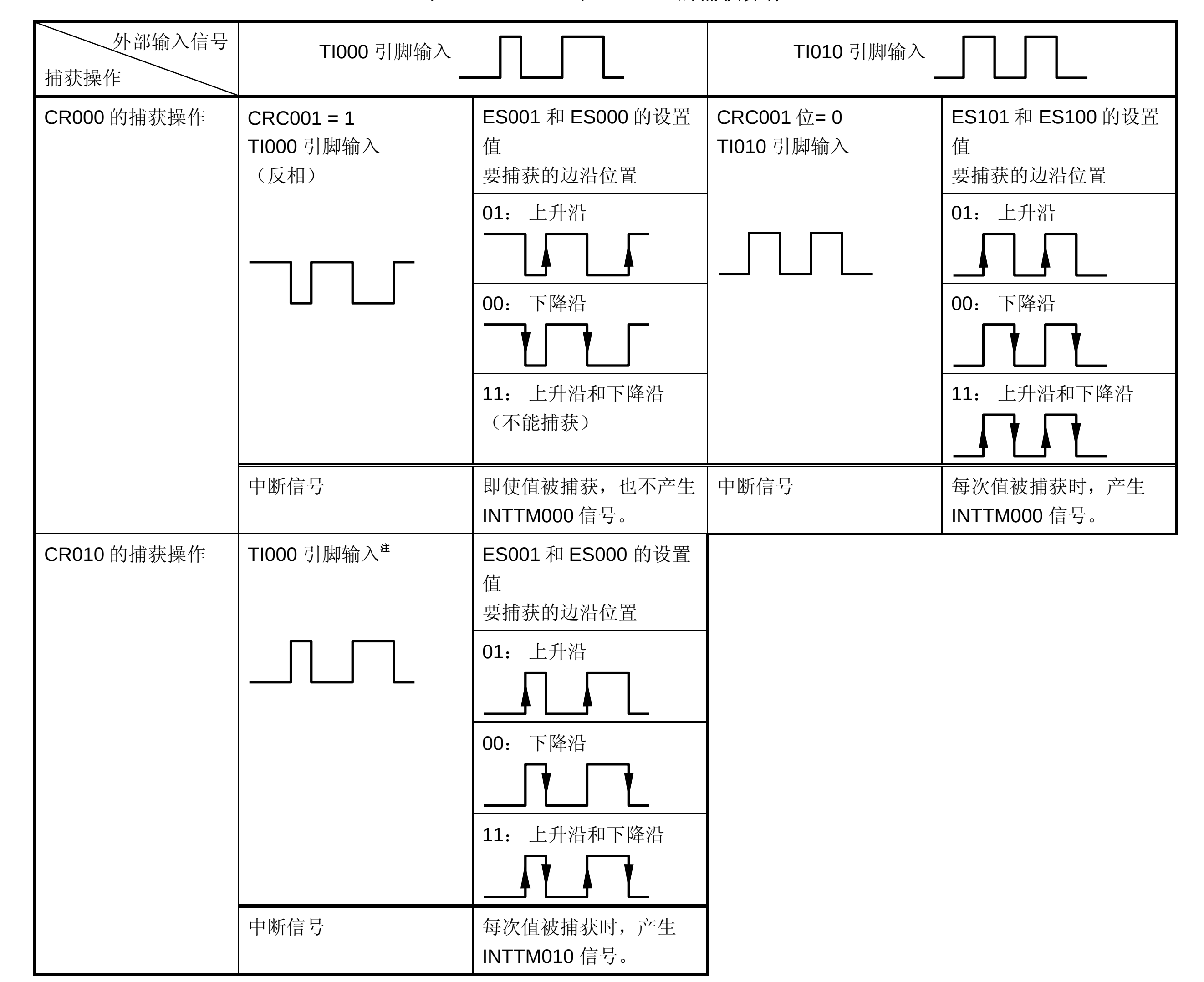
- 当定时器计数器由于溢出被清除时
- 当定时器计数器由于 TI000 引脚有效边沿（有效边沿输入 TI000 引脚时进入清零&启动模式）被清除时
- 当定时器计数器由于比较匹配（TM00 与 CR000 匹配（CR000 = 非 0000H，CR010 = 0000H）时进入清零&启动模式）被清除时



备注　**1.** N：CR000 寄存器设置值，M：CR010 寄存器设置值

2. TMC003 和 TMC002 的详情参见 **6.3（1）16 位定时器模式控制寄存器 00（TMC00）**。

表 6-2. CR000 和 CR010 的捕获操作



外部输入信号 / 捕获操作	TI000 引脚输入		TI010 引脚输入	
CR000 的捕获操作	CRC001 = 1 TI000 引脚输入 （反相）	ES001 和 ES000 的设置值 要捕获的边沿位置	CRC001 位= 0 TI010 引脚输入	ES101 和 ES100 的设置值 要捕获的边沿位置
		01： 上升沿		01： 上升沿
		00： 下降沿		00： 下降沿
		11： 上升沿和下降沿 （不能捕获）		11： 上升沿和下降沿
	中断信号	即使值被捕获，也不产生 INTTM000 信号。	中断信号	每次值被捕获时，产生 INTTM000 信号。
CR010 的捕获操作	TI000 引脚输入[注]	ES001 和 ES000 的设置值 要捕获的边沿位置		
		01： 上升沿		
		00： 下降沿		
		11： 上升沿和下降沿		
	中断信号	每次值被捕获时，产生 INTTM010 信号。		

注　　CR010 的捕获操作不受 CRC001 位设置的影响。

注意事项　如果要使用输入到 **TI000** 引脚的反向脉冲来捕获 **TM00** 的计数值，并将之捕获到 **CR000**，则在捕获到计数值后，不产生中断请求信号（**INTTM000**）。如果在这个操作期间检测到 **TI010** 的有效边沿，则不会执行捕获操作，但会产生 **INTTM000** 信号作为外部中断信号。不使用外部中断时，屏蔽 **INTTM000** 信号。

备注　　CRC001：参见 **6.3**（**2**） **捕获/**比较控制寄存器 **00**（**CRC00**）。
ES101，ES100，ES001，ES000： 参见 **6.3**（**4**） 预分频模式寄存器 **00**（**PRM00**）。

6.3 控制 16 位定时器/事件计数器 00 的寄存器

用于控制 16 位定时器/事件计数器 00 的寄存器如下所示。

- 16 位定时器模式控制寄存器 00（TMC00）
- 捕获/比较控制寄存器 00（CRC00）
- 16 位定时器输出控制寄存器 00（TOC00）
- 预分频模式寄存器 00（PRM00）
- 输入切换控制寄存器（ISC）
- 端口模式寄存器 3（PM3）
- 端口寄存器 3（P3）

（1） 16 位定时器模式控制寄存器 00（TMC00）

TMC00 是 8 位寄存器，用于设置 16 位定时器/事件计数器 00 的操作模式、TM00 清零模式和输出时序，并检测溢出。

操作期间（当 TMC003 与 TMC002 不等于 00）禁止重写 TMC00。但是，当 TMC003 与 TMC002 被清除为 00 时（停止操作），当 OVF00 被清除为 0 时，可以改变它的值。

可以由 1 位或 8 位存储器操作指令来设置 TMC00。

复位信号的产生会将 TMC00 清除为 00H。

注意事项 当 TMC002 与 TMC003 被设置为 00（操作停止模式）以外的值时，16 位定时器/事件计数器 00 开始操作。TMC002 和 TMC003 被设置为 00 时，可以停止操作。

图 6-5. 16 位定时器模式控制寄存器 00（TMC00）的格式

地址： FFBAH　　复位后： 00H　　R/W

符号	7	6	5	4	3	2	1	<0>
TMC00	0	0	0	0	TMC003	TMC002	TMC001	OVF00

TMC003	TMC002	16 位定时器/事件计数器 00 的操作使能
0	0	禁止 16 位定时器/事件计数器 00 操作。停止操作时钟的供应。清除 16 位定时器计数器 00（TM00）。
0	1	自由运行定时器模式
1	0	有效边沿输入[注]TI000引脚时进入清零&启动模式
1	1	TM00和CR000匹配时进入清零&启动模式

TMC001	反转定时器输出（TO00）的条件
0	• TM00与CR000匹配，或TM00与CR010匹配
1	• TM00与CR000匹配，或TM00与CR010匹配 • TI000引脚有效边沿的触发输入

OVF00	TM00溢出标志
清除（0）	将OVF00清除为0，或TMC003与TMC002 = 00
置位（1）	发生溢出
在所有操作模式下（自由运行定时器模式、清零&启动模式（通过有效边沿输入TI000引脚进入）和清零&启动模式（在TM00 与CR000匹配时进入）），当TM00的值由FFFFH变为0000H时，OVF00标志位将被设置为1。 也可以将1写入OVF00进行设置。	

注　通过预分频器模式寄存器 00（PRM00）的第 5 位和第 4 位（ES001，ES000）设置 TI000 引脚的有效边沿。

（2） 捕获/比较控制寄存器 00（CRC00）

CRC00 寄存器用于控制 CR000 和 CR010 的操作。

操作期间（TMC003 与 TMC002 不等于 00 时）禁止改变 CRC00。

可以由 1 位或 8 位存储器操作指令来设置 CRC00。

复位信号的产生会将 CRC00 清除为 00H。

图 6-6. 捕获/比较控制寄存器 00（CRC00）的格式

地址： FFBCH 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
CRC00	0	0	0	0	0	CRC002	CRC001	CRC000

CRC002	CR010 操作模式的选择
0	用作比较寄存器
1	用作捕获寄存器

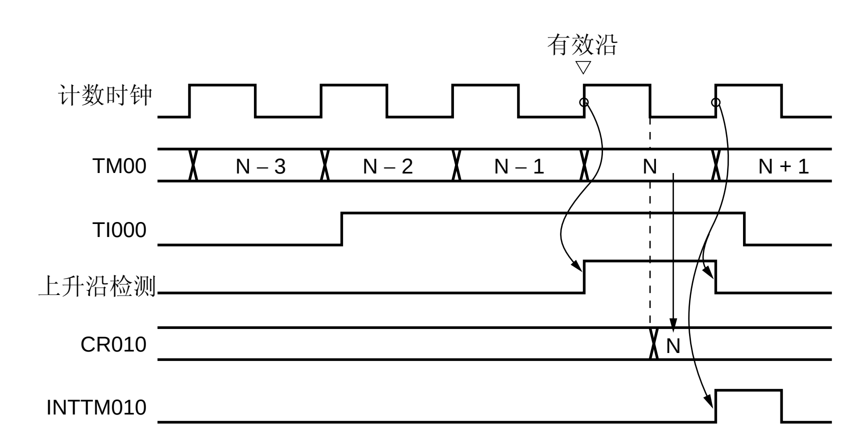
CRC001	CR000 捕获触发选择
0	在 TI010 引脚的有效边沿捕获
1	在 TI000 引脚的有效边沿反相[注] 捕获
由 PRM00 设置 TI010 与 TI000 引脚的有效边沿。 如果在 CRC001 为 1 时将 ES001 和 ES000 设置为 11（两种边沿），则无法检测 TI000 引脚的有效边沿	

CRC000	CR000 操作模式的选择
0	用作比较寄存器
1	用作捕获寄存器
如果 TMC003 与 TMC002 被设置为 11（TM00 与 CR000 匹配时进入清零&启动模式），则必须将 CRC000 设置为 0。	

注 当检测到 TI000 的有效边沿时，不会执行捕获操作，但是产生 INTTM000 信号作为外部中断信号。

注意事项 为确保捕获操作正确执行，捕获触发需要的脉冲应大于计数时钟的两个时钟周期，该计数时钟由预分频器模式寄存器 00（PRM00）选择。

图 6-7. CR010 捕获操作的示例（指定上升沿有效）



（3） 16 位定时器输出控制寄存器 00（TOC00）

TOC00 是 8 位寄存器，用于控制 TO00 输出。

仅当 OSPT00 操作（当 TMC003 与 TMC002 不等于 00 时）时才能重写 TOC00。操作期间禁止重写其它位。

但是在定时器操作期间可以重写 TOC004，作为重写 CR010 的一种方法（参见 **6.5.1 TM00 操作期间重写 CR010**）。

可以由 1 位或 8 位存储器操作指令来设置 TOC00。

复位信号的产生会将 TOC00 清除为 00H。

注意事项　请确保按以下过程设置 TOC00。

<1> 设置 TOC004 和 TOC001 为 1。

<2> 仅设置 TOE00 = 1。

<3> 设置 LVS00 或 LVR00 为 1。

图 6-8. 16 位定时器输出控制寄存器 00（TOC00）的格式

地址： FFBDH 复位后：00H R/W

符号	7	<6>	<5>	4	<3>	<2>	1	<0>
TOC00	0	OSPT00	OSPE00	TOC004	LVS00	LVR00	TOC001	TOE00

OSPT00	通过软件触发单次触发脉冲输出
0	–
1	单次触发脉冲输出
读取该位获得的值总是为“0”。在其他所有模式下（单次触发脉冲输出模式除外），不要将该位设置为 1。如果该位被设置为 1，TM00 被清除并启动。	

OSPE00	单次触发脉冲输出操作控制
0	连续脉冲输出模式
1	单次触发脉冲输出
在自由运行定时器模式或清零&启动模式（通过有效边沿输入 TI000 引脚进入）下，单次触发脉冲输出可以正确操作。 在 TM00 与 CR000 匹配时清零&启动的模式下，不能输出单次触发脉冲。	

TOC004	CR010 与 TM00 匹配时 TO00 引脚的输出控制
0	禁止反转操作
1	使能反转操作
即使 TOC004 = 0，也会产生中断信号（INTTM010）。	

LVS00	LVR00	TO00 输出状态的设置
0	0	无变化
0	1	TO00 引脚输出的初始值为低电平（TO00 输出被清除为 0）
1	0	TO00 引脚输出的初始值为高电平（TO00 输出被设置为 1）
1	1	禁止设置

- LVS00 与 LVR00 可以用来设置 TO00 输出电平的初始值。如果不必设置初始值，则保持 LVS00 与 LVR00 为 00。
- 当 TOE00 = 1 时，请确保设置 LVS00 与 LVR00。
  禁止同时将 LVS00、LVR00 和 TOE00 设置为 1。
- LVS00 与 LVR00 是触发位。通过将这些位设置为 1，可以设置 TO00 输出电平的初始值。即使将这些位清除为 0，TO00 输出也不会受到影响。
- 读取 LVS00 与 LVR00 获得的值总是为 0。
- 如需了解如何设置 LVS00 与 LVR00，参见 **6.5.2 LVS00 与 LVR00 的设置。**
- 除 TO00 输出之外，TO00/TI010/P34/TI52/RTC1HZ/INTP1 引脚实际的输出由 PM34 和 P34 决定。

TOC001	CR000 与 TM00 匹配时 TO00 的输出控制
0	禁止反转操作
1	使能反转操作
即使 TOC001 = 0，也产生中断信号（INTTM000）	

TOE00	TO00 输出控制
0	禁止输出（TO00 输出固定为低电平）
1	使能输出

（4） 预分频模式寄存器 00（PRM00）

PRM00 寄存器用于设置 TM00 的计数时钟以及 TI000 和 TI010 引脚输入的有效边沿。

操作期间（当 TMC003 与 TMC002 不等于 00 时）禁止重写 PRM00。

可由 1 位或 8 位存储器操作指令来设置 PRM00。

复位信号的产生会将 PRM00 清除为 00H。

注意事项 1. 当 PRM001 与 PRM000 为 11 时（指定 TI000 引脚的有效边沿作为计数时钟），不要进行以下设置。

- **有效边沿输入 TI000 引脚时进入清零&启动模式**
- **设置 TI000 引脚作为捕获触发**

2. 如果在 TI000 或 TI010 引脚为高电平且指定 TI000 或 TI010 引脚的有效边沿为上升沿或双沿时，使能 16 位定时器/事件计数器 00 的操作，TI000 或 TI010 引脚的高电平将被检测认为是上升沿。当 TI000 或 TI010 引脚被上拉时要注意。但是，一旦定时器操作已经被停止然后再次使能时，不检测上升沿。

3. P34 引脚不能同时用作 TI010 有效边沿和定时器输出（TO00）。请选择其中一个功能。

图 6-9. 预分频模式寄存器 00（PRM00）的格式

地址： FFBBH 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
PRM00	ES101	ES100	ES001	ES000	0	PRM002	PRM001	PRM000

ES101	ES100	TI010 引脚有效边沿选择
0	0	下降沿
0	1	上升沿
1	0	禁止设置
1	1	上升沿和下降沿双沿

ES001	ES000	TI000 引脚有效边沿选择
0	0	下降沿
0	1	上升沿
1	0	禁止设置
1	1	上升沿和下降沿双沿

PRM002	PRM001	PRM000	计数时钟选择[注 1]			
				f_{PRS} = 2 MHz	f_{PRS} = 5 MHz	f_{PRS} = 10 MHz
0	0	0	f_{PRS}[注 2]	2 MHz	5 MHz	10 MHz
0	0	1	$f_{PRS}/2$	1 MHz	2.5 MHz	5 MHz
0	1	0	$f_{PRS}/2^2$	500 kHz	1.25 MHz	2.5 MHz
0	1	1	$f_{PRS}/2^4$	1.25 MHz	2.5 MHz	625 kHz
1	0	0	$f_{PRS}/2^8$	7.81 kHz	19.53 kHz	39.06 kHz
1	0	1	f_{SUB}	32.768 kHz		
1	1	0	TI000 有效边沿[注 3]			
1	1	1	TM52 输出			

注 **1.** 如果外设硬件时钟（f_{PRS}）是高速系统时钟（f_{XH}）（XSEL = 1），根据供电电压的不同，f_{PRS} 的工作频率也不同。
- V_{DD} = 2.7 至 5.5 V： $f_{PRS} \leq 10$ MHz
- V_{DD} = 1.8 至 2.7 V： $f_{PRS} \leq 5$ MHz

2. 如果外设硬件时钟（f_{PRS}）是内部高速振荡时钟（f_{RH}）（XSEL = 0），当 1.8 V ≤ V_{DD} < 2.7 V 时，PRM002 = PRM001 = PRM000 = 0（计数时钟： f_{PRS}）的设置被禁止。

3. 来自 TI000 引脚的外部时钟的一个脉冲宽度要求大于外设硬件时钟（f_{PRS}）的两个周期。

注意事项 在脉冲宽度测量期间，不要选择 TI000 的有效边沿作为计数时钟。

备注 **1.** 通过设置 PRM002，PRM001，PRM000 = 1，1，1，8 位定时器/事件计数器 52（TM52）的输出可以选择作为 TM00 计数时钟。根据 TM52 计数时钟和比较寄存器的设置值，可设置任意频率作为 16 位定时器（TM00）计数时钟。

2. f_{PRS}： 外设硬件时钟频率
f_{SUB}： 副系统时钟频率

(5) 输入切换控制寄存器（ISC）

将 ISC1 设置为 0，TI000 的输入源成为来自 P33/TI000 引脚的输入信号。
可由 1 位或 8 位存储器操作指令来设置 ISC 。
复位信号的产生会将 ISC 清除为 00H。

图 6-10. 输入切换控制寄存器（ISC）的格式

地址： FF4FH 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
ISC	0	0	ICS5	ICS4	ICS3	ICS2	ICS1	ICS0

ICS5	ICS4	TxD6，RxD6 输入源选择
0	0	TxD6：P112，RxD6： P113
1	0	TxD6：P13，RxD6： P12
其他值		禁止设置

ISC3	RxD6/P113 输入使能/禁止
0	RxD6/P113 输入禁止
1	RxD6/P113 输入使能

ISC2	TI52 输入源控制
0	未使能 TI52 输入的控制（P34）
1	使能 TI52 输入的控制（P34）[注]

ISC1	TI000 输入源选择
0	TI000（P33）
1	RxD6（P12 或 P113[注 2]）

ISC0	INTP0 输入源选择
0	INTP0（P120）
1	RxD6（P12 或 P113[注 2]）

<R> 注 1. TI52 的输入由 TOH2 输出信号控制。
2. 通过 ISC5 和 ISC4 来选择。

（6） 端口模式寄存器 3（PM3）

该寄存器按位设置端口 3 为输入/输出模式。

如果 P34/TI52/TI010/TO00/RTC1HZ/INTP1 引脚用作定时器输出，则需要将 PM34 以及 P34 的输出锁存器为 0。

如果 P33/TI000/RTCDIV/RTCCL/BUZ/INTP2 和 P34/TI52/TI010/TO00/RTC1HZ/INTP1 引脚用作定时器输入，则需要将 PM33 和 PM34 设置为 1。此时，P33 和 P34 的输出锁存器可以为 0 或为 1。

可由 1 位或 8 位存储器操作指令来设置 PM3 。

复位信号的产生会将 PM3 设置为 FFH。

图 6-11. 端口模式寄存器 3（PM3）的格式

地址：FF23H　复位后：FFH　R/W

符号	7	6	5	4	3	2	1	0
PM3	1	1	1	PM34	PM33	PM32	PM31	PM30

PM3n	P3n 引脚 I/O 模式选择 (n = 0 至 4)
0	输出模式 (输出缓冲器打开)
1	输入模式 (输出缓冲器关闭)

6.4 16 位定时器/事件计数器 00 的操作

6.4.1 间隔定时器操作

如果 16 位定时器模式控制寄存器（TMC00）的第 3 位和第 2 位（TMC003 和 TMC002）为 11（TM00 与 CR000 匹配时进入清零&启动模式），则随着计数时钟同步启动计数操作。

当 TM00 随后的值与 CR000 值匹配时，将 TM00 清除为 0000H 并产生一个匹配中断信号（INTTM000）。这个 INTTM000 信号使能 TM00 作为间隔定时器操作。

备注 **1.** I/O 引脚设置的详细说明，参见 **6.3（6） 端口模式 寄存器 3（PM3）**。
2. 要了解如何使能 INTTM000 中断，参见 **第二十章 中断功能**。

图 6-12. 间隔定时器操作的框图

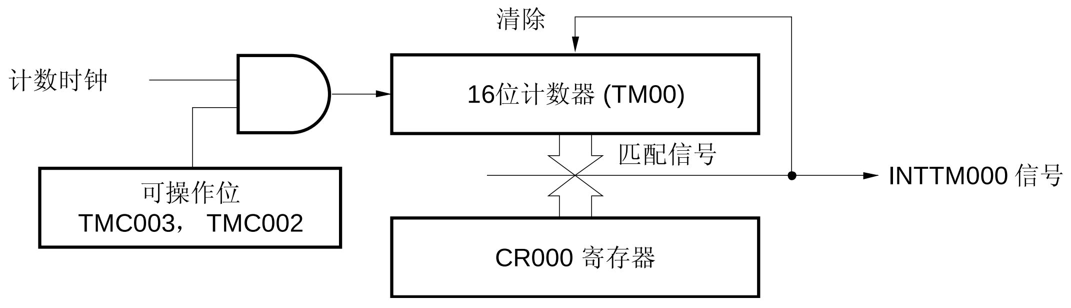


图 6-13. 间隔定时器操作的基本时序示例

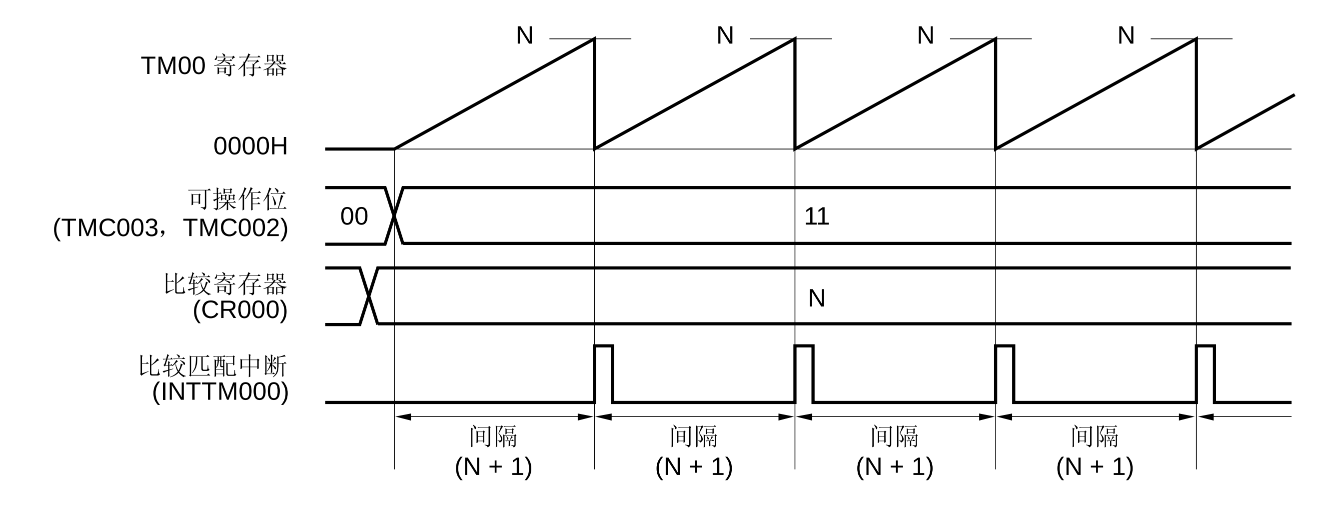
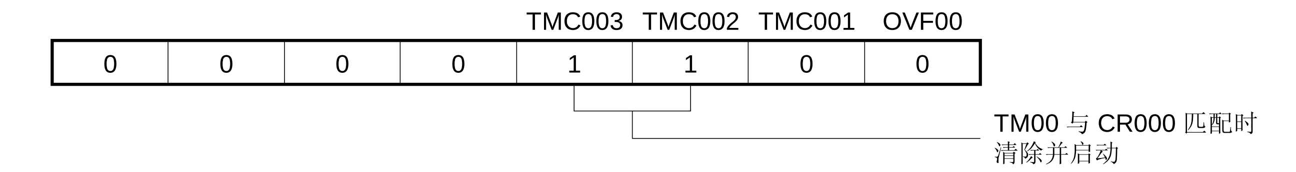


图 6-14. 间隔定时器操作的寄存器设置示例

（a） **16 位定时器模式控制寄存器 00（TMC00）**



（b） **捕获/比较控制寄存器 00（CRC00）**

					CRC002	CRC001	CRC000
0	0	0	0	0	0	0	0

CRC000 = 0：CR000 用作比较寄存器

（c） **16 位定时器输出控制寄存器 00（TOC00）**

	OSPT00	OSPE00	TOC004	LVS00	LVR00	TOC001	TOE00
0	0	0	0	0	0	0	0

（d） **预分频模式寄存器 00（PRM00）**

ES101	ES100	ES001	ES000	3	PRM002	PRM001	PRM000
0	0	0	0	0	0/1	0/1	0/1

PRM002～PRM000：选择计数时钟

（e） **16 位定时器计数器 00（TM00）**

通过读取 TM00，获得计数值。

（f） **16 位捕获/比较寄存器 000（CR000）**

如果将 M 设置到 CR000，则间隔时间如下。

- 间隔时间 =（M + 1）× 计数时钟周期

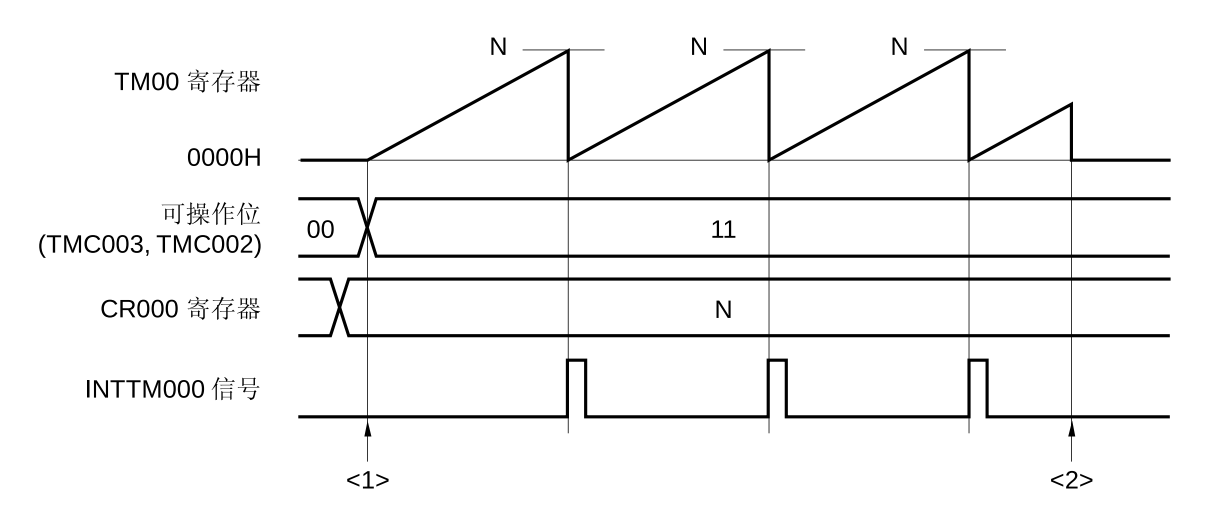
禁止将 CR000 设置为 0000H。

（g） **16 位捕获/比较寄存器 010（CR010）**

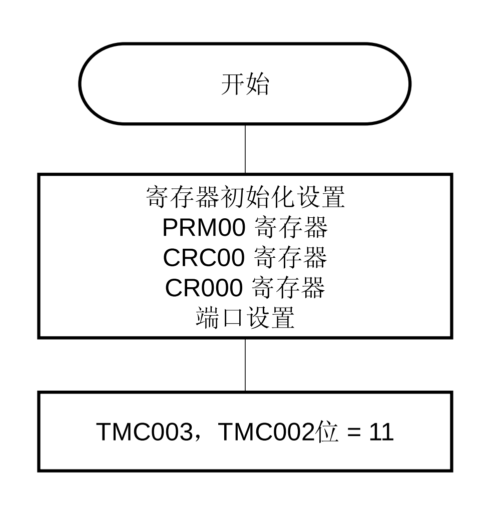
通常 CR010 不用作间隔定时器功能。但当 CR010 的值与 TM00 的值匹配时，会产生比较匹配中断（INTTM010）。

因此，使用中断屏蔽标志（TMMK010）屏蔽该中断请求。

图 6-15. 间隔定时器功能软件处理的示例



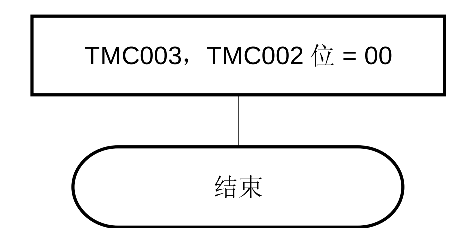
<1> 计数操作启动流程



在将 TMC003 和 TMC002 设置为 11之前，
先对这些寄存器进行初始化设置。

开始计数操作

<2> 计数操作停止流程



通过将 TMC003 和 TMC002 清零，
对计数器进行初始化，并停止计数。

6.4.2 方波输出操作

当 16 位定时器/事件计数器 00 用作间隔定时器时（参见 **6.4.1**），将 16 位定时器输出控制寄存器 00（TOC00）设置为 03H，可以从 TO00 引脚输出一个方波。

当 TMC003 与 TMC002 被设置为 11 时（TM00 与 CR000 匹配时进入计数清零&启动模式），随着计数时钟同步启动计数操作。

当 TM00 随后的值与 CR000 的值匹配时，将 TM00 清除为 0000H、产生中断信号（INTTM000）并反转 TO00 的输出。被反转的 TO00 输出以固定间隔使能 TO00 输出一个方波。

备注 **1.** 如需了解 I/O 引脚的设置，可参见 **6.3（6） 端口模式寄存器 3（PM3）**。

2. 要了解如何使能 INTTM000 信号中断，参见 **第二十章 中断功能**。

图 6-16. 方波输出操作框图

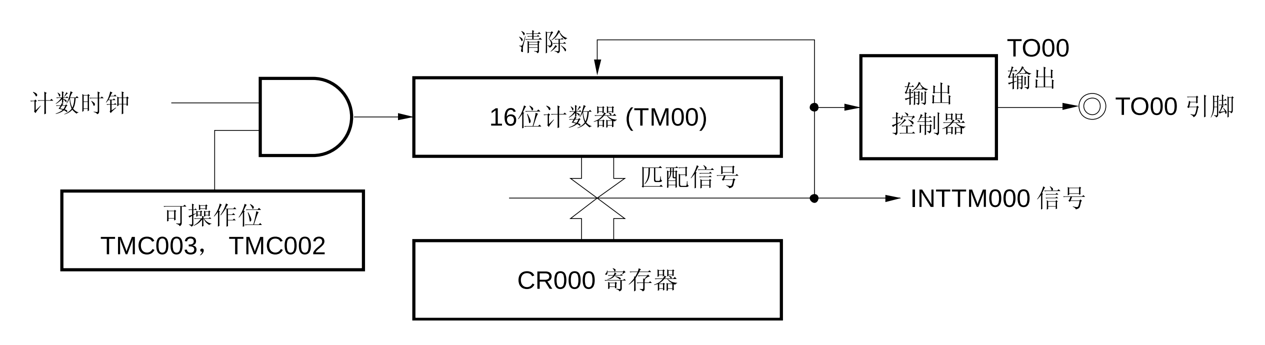


图 6-17. 方波输出操作的基本时序示例

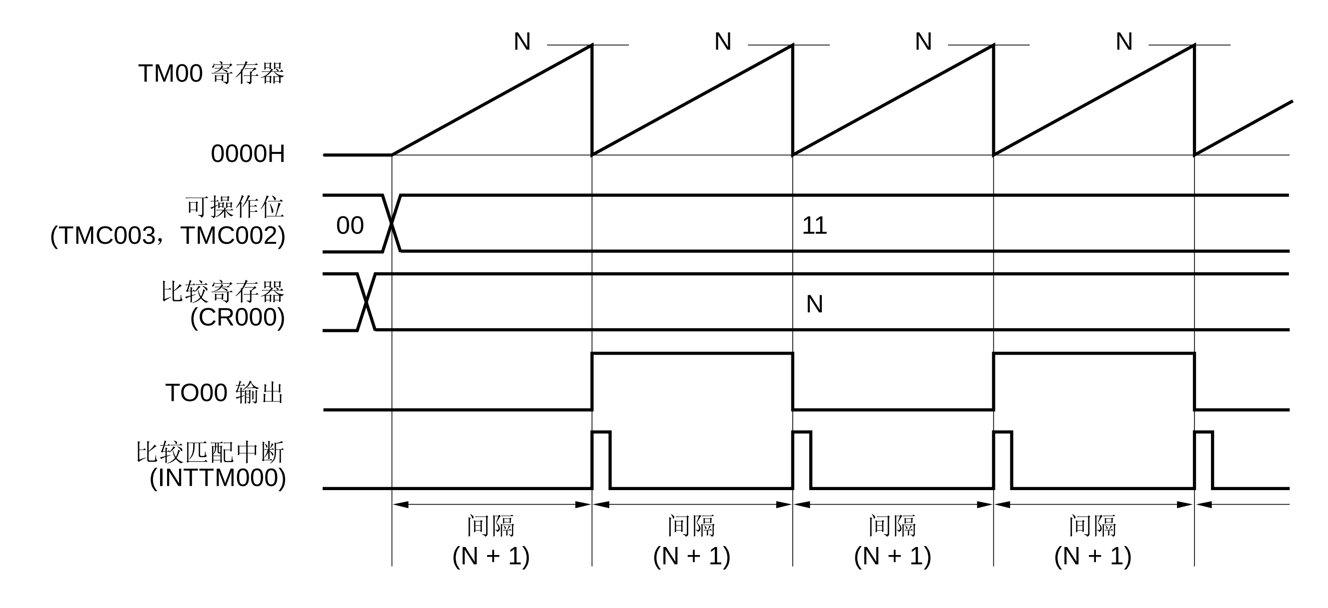
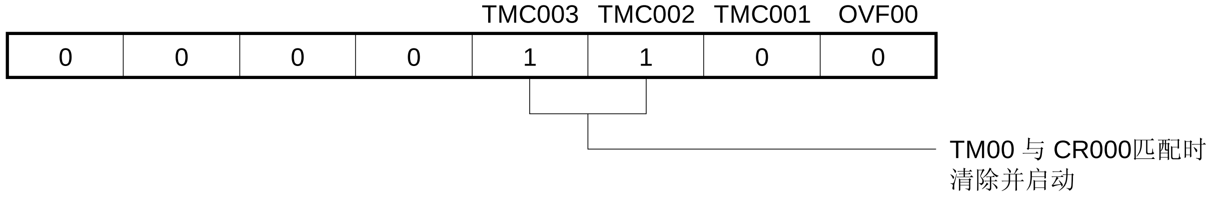


图 **6-18.** 方波输出操作的寄存器设置示例

（**a**） **16** 位定时器模式控制寄存器 **00**（**TMC00**）

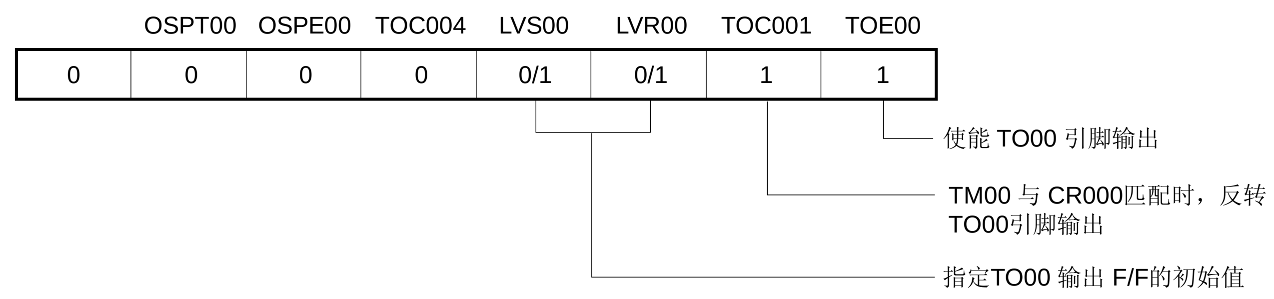


（**b**） 捕获**/**比较控制寄存器 **00**（**CRC00**）

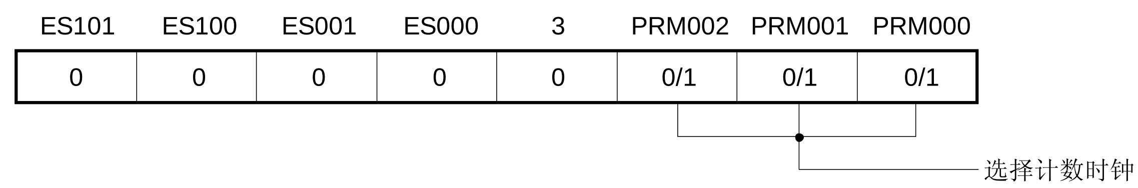
					CRC002	CRC001	CRC000
0	0	0	0	0	0	0	0

CRC000：CR000 用作比较寄存器

（**c**） **16** 位定时器输出控制寄存器 **00**（**TOC00**）



（**d**） 预分频模式寄存器 **00**（**PRM00**）



（**e**） **16** 位定时器计数器 **00**（**TM00**）

通过读取 TM00，获得计数值。

（**f**） **16** 位 捕获**/**比较寄存器 **000**（**CR000**）

如果将 M 设置到 CR000，则间隔时间表示如下。

- 方波频率 = 1 / [2 ×（M + 1）× 计数时钟周期]

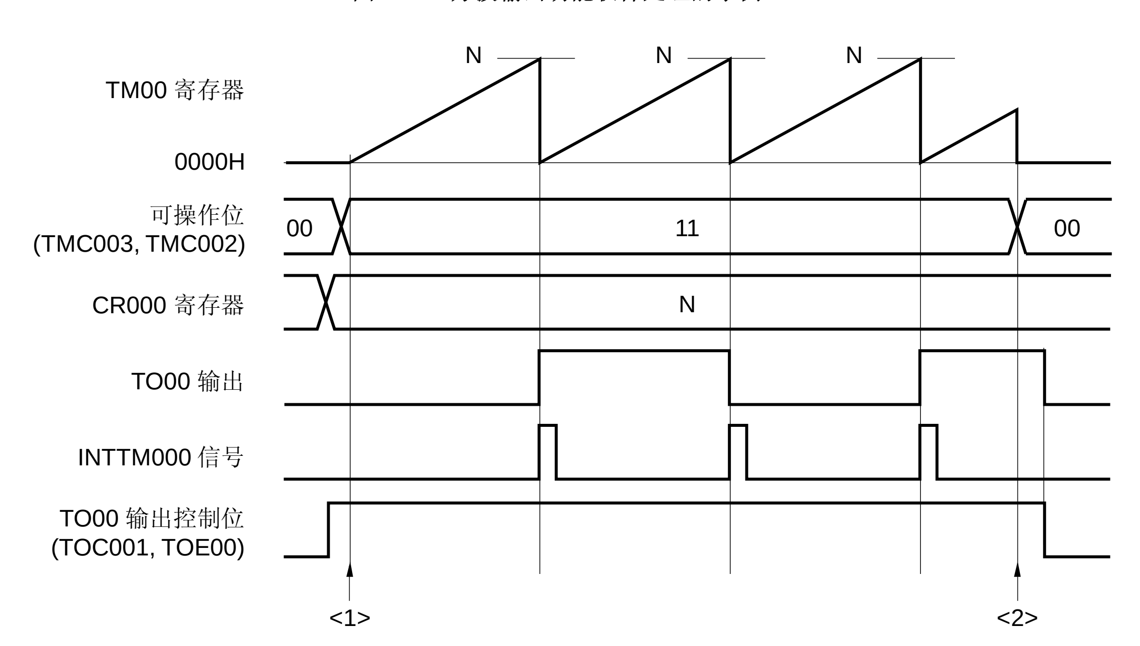
禁止将 CR000 设置为 0000H。

（**g**） **16** 位捕获**/**比较寄存器 **010**（**CR010**）

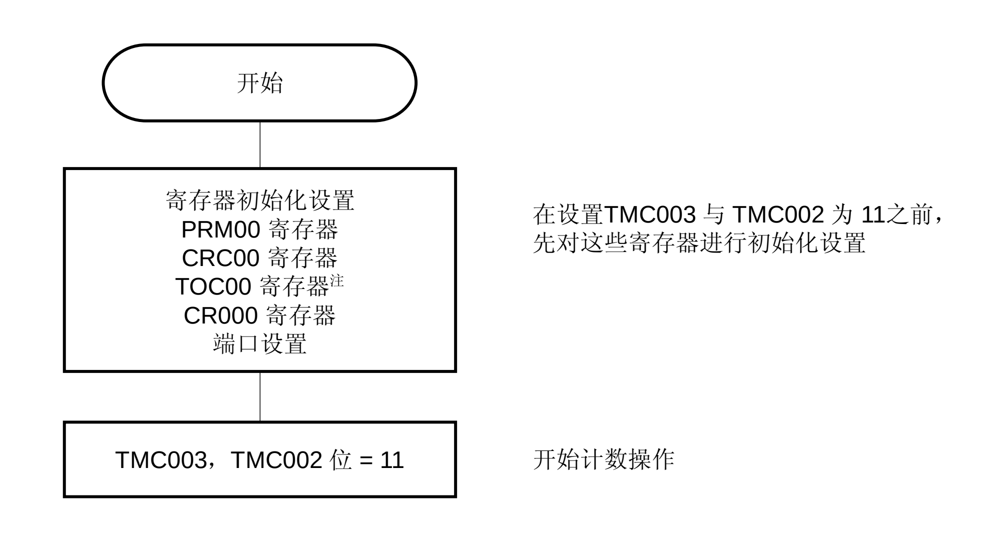
通常 CR010 不用作方波输出功能。但当 CR010 的值与 TM00 的值匹配时，会产生比较匹配中断（INTTM010）。

因此，使用中断屏蔽标志（TMMK010）屏蔽该中断请求。

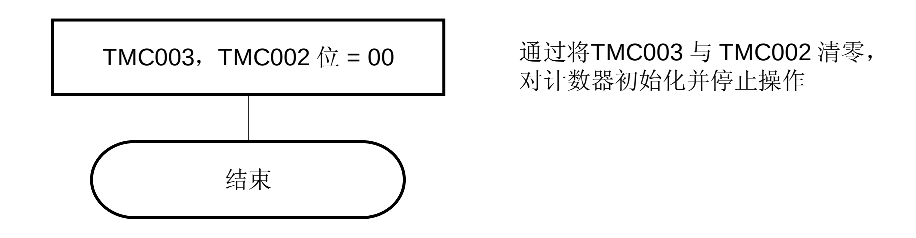
图 6-19. 方波输出功能软件处理的示例



<1> 计数操作启动流程



<2>计数操作停止流程



注　　设置 TOC00 时必须特别注意。详情参见 **6.3（3）16 位定时器输出控制寄存器 00（TOC00）**。

6.4.3 外部事件计数器操作

当预分频器模式寄存器 00（PRM00）的第 1 位和第 0 位（PRM001 和 PRM000）被设置为 11（TI000 引脚的有效边沿向上计数）且 16 位定时器模式控制寄存器 00（TMC00）的第 3 位和第 2 位（TMC003 和 TMC002）被设置为 11 时，对外部事件输入的有效边沿进行计数，并产生用于指示 TM00 和 CR000 匹配的匹配中断信号（INTTM000）。

要输入外部事件，可以使用 TI000 引脚。因此，定时器/事件计数器不能在通过 TI000 引脚有效边沿进入清零&启动（当 TMC003 与 TMC002 = 10 时）的模式下用作外部事件计数器。

INTTM000 信号按下列时序产生。

- 产生 INTTM000 信号的时序（第 2 次或随后）
  = 外部事件输入的有效边沿检测次数 ×（CR000 的设置值 + 1）

但是，在定时器/事件计数器开始操作后，将按以下时序立即产生第 1 次匹配中断。

- 产生 INTTM000 信号的时序（仅在第 1 次产生）
  = 外部事件输入的有效边沿检测次数×（CR000 的设置值 + 2）

要检测有效边沿，则应在 f_{PRS} 时钟周期期间对输入到 TI000 引脚的信号进行采样。在连续检测到两次时才认为检测到有效边沿，于是，可以消除一个窄脉冲宽度的噪声。

备注 **1.** 如需了解 I/O 引脚的设置，参见 **6.3（6） 端口模式寄存器 3（PM3）**。
2. 要了解如何使能 INTTM000 信号中断，参见**第二十章 中断功能**。

图 6-20. 外部事件计数器操作的框图

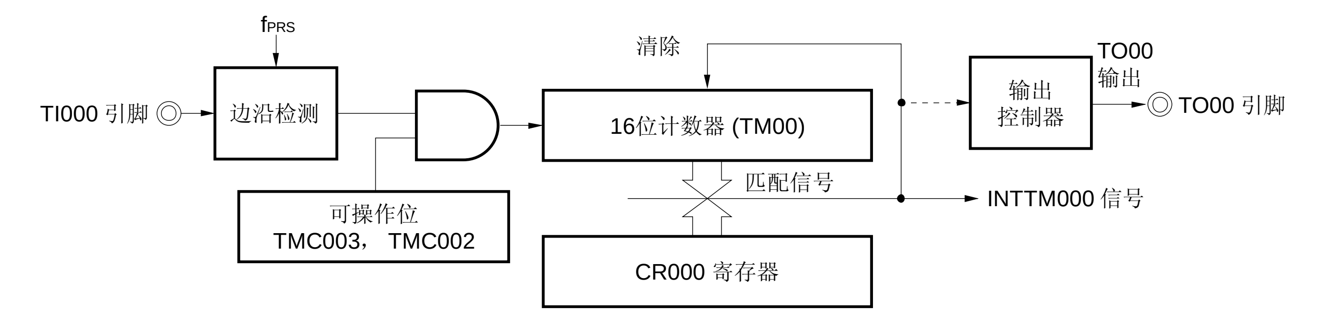
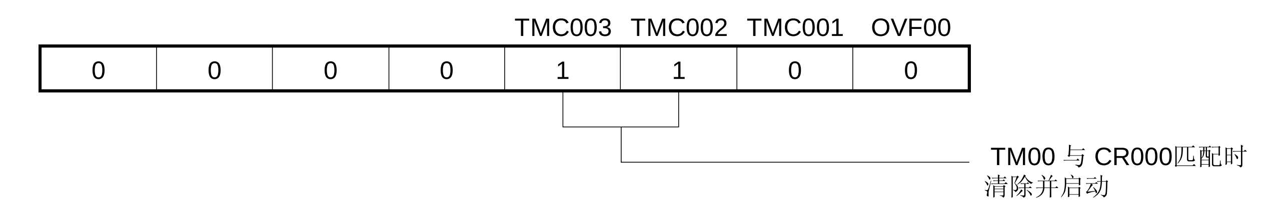
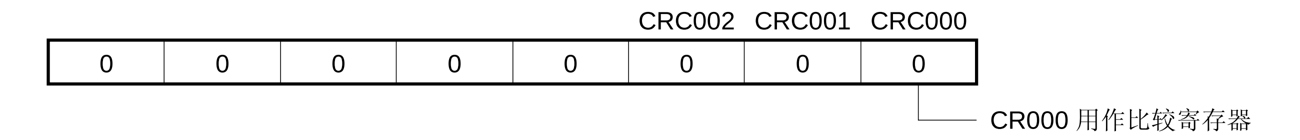


图 6-21. 外部事件计数器模式下寄存器设置示例（1/2）

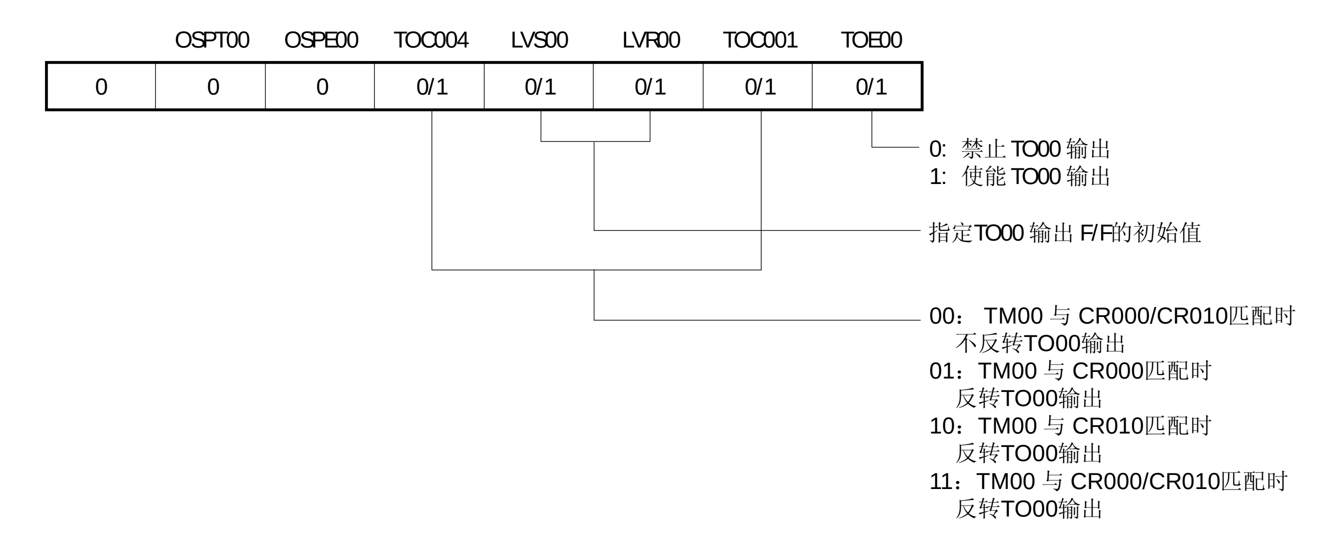
（a） 16 位定时器模式控制寄存器 00（TMC00）



（b） 捕获/比较控制寄存器 00（CRC00）



（c） 16 位定时器输出控制寄存器 00（TOC00）



（d） 预分频模式寄存器 00（PRM00）



图 6-21. 外部事件计数器模式下寄存器设置示例（2/2）

（e） 16 位定时器计数器 00（TM00）

通过读取 TM00，获得计数值。

（f） 16 位捕获/比较寄存器 000（CR000）

如果将 M 设置到 CR000，当外部事件达到（M+1）时产生中断信号（INTTM000）。

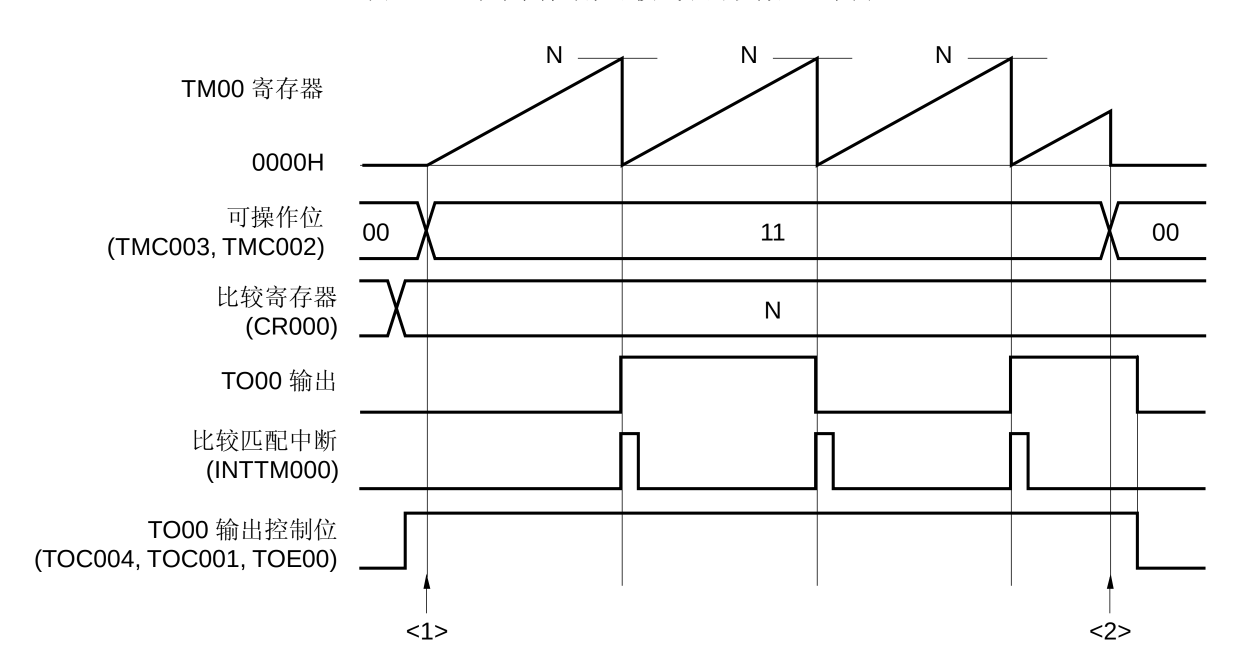
禁止将 CR000 设置为 0000H。

（g） 16 位捕获/比较寄存器 010（CR010）

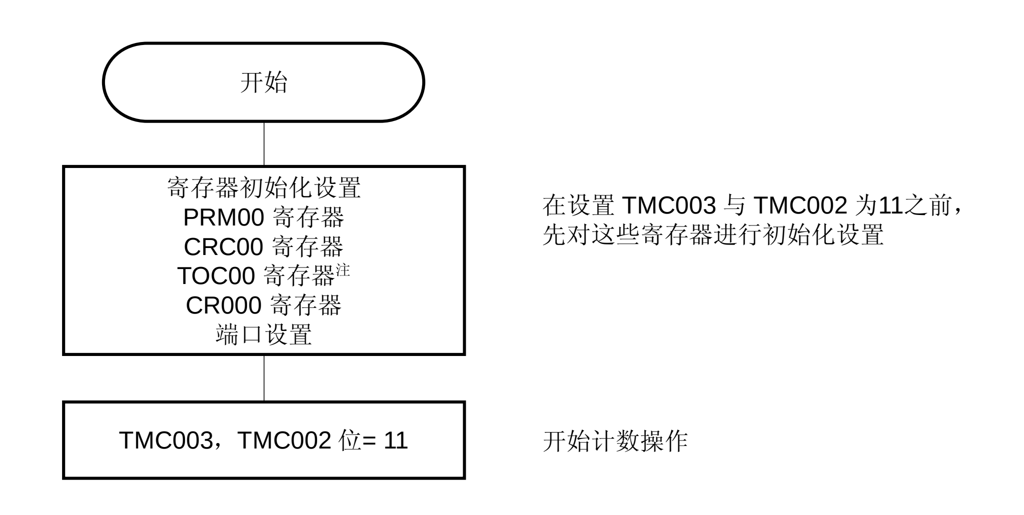
通常 CR010 不用于外部事件计数器模式。但当 CR010 的值与 TM00 的值匹配时，会产生比较匹配中断（INTTM010）。

因此，使用中断屏蔽标志（TMMK010）屏蔽该中断请求。

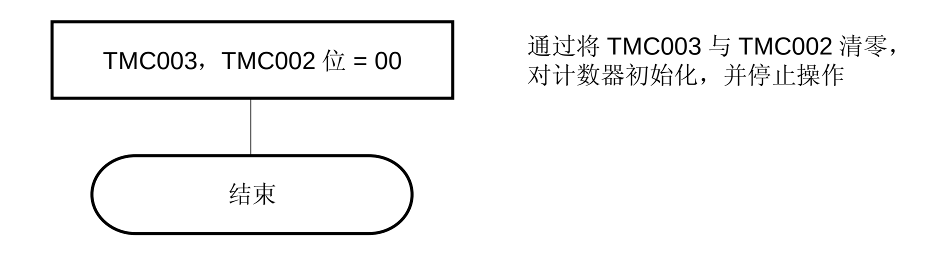
图 6-22. 外部事件计数器模式下的软件处理示例



<1> 计数操作启动流程



<2> 计数操作停止流程



注　　设置 TOC00 时必须特别注意。详情参见 **6.3（3）16 位定时器输出控制寄存器 00（TOC00）**。

6.4.4 通过TI000 引脚有效边沿输入进入的清零&启动模式下的操作

当 16 位定时器模式控制寄存器 00（TMC00）的第 3 位和第 2 位（TMC003 和 TMC002）被设置为 10（通过输入 TI000 引脚的有效边沿进入清零&启动模式）且向定时器/事件计数器提供计数时钟（由 PRM00 设置）时，TM00 开始向上计数。如果在计数操作期间检测到 TI000 引脚的有效边沿，则将 TM00 清除为 0000H 并再次开始计数。如果没有检测到 TI000 引脚的有效边沿，则 TM00 溢出并继续计数。

TI000 引脚的有效边沿是清除 TM00 的一个原因。在操作启动后，不要立即启动计数器。

CR000 和 CR010 可用作比较寄存器和捕获寄存器。

（a） CR000 和 CR010 用作比较寄存器

当 TM00 的值与 CR000、CR010 的值匹配时，产生信号 INTTM000 与 INTTM010。

（b） CR000 和 CR010 用作捕获寄存器

当有效边沿输入 TI010 引脚时（或者当有效边沿的反相输入到 TI000 引脚时），TM00 的计数值被捕获到 CR000，并产生 INTTM000 信号。

当有效边沿输入到 TI000 引脚时，TM00 的计数值被捕获到 CR010，并产生 INTTM010 信号。只要捕获到计数值，计数器即被清除为 0000H。

注意事项 **不要将 TI000 引脚的有效边沿设置为计数时钟（PRM002，PRM001 和 PRM000 = 110）。当 PRM002，PRM001 和 PRM000 = 110 时，TM00 被清除。**

备注 **1.** 如需了解 I/O 引脚的设置，参见 **6.3（6） 端口模式寄存器 3（PM3）**。
2. 要了解如何使能 INTTM000 信号中断，参见**第二十章 中断功能**。

（1） 通过输入 TI000 引脚的有效边沿进入清零&启动模式下的操作
（CR000：比较寄存器，CR010：比较寄存器）

图 6-23. 通过输入 TI000 引脚的有效边沿进入清零&启动模式的框图
（CR000： 比较寄存器，CR010： 比较寄存器）

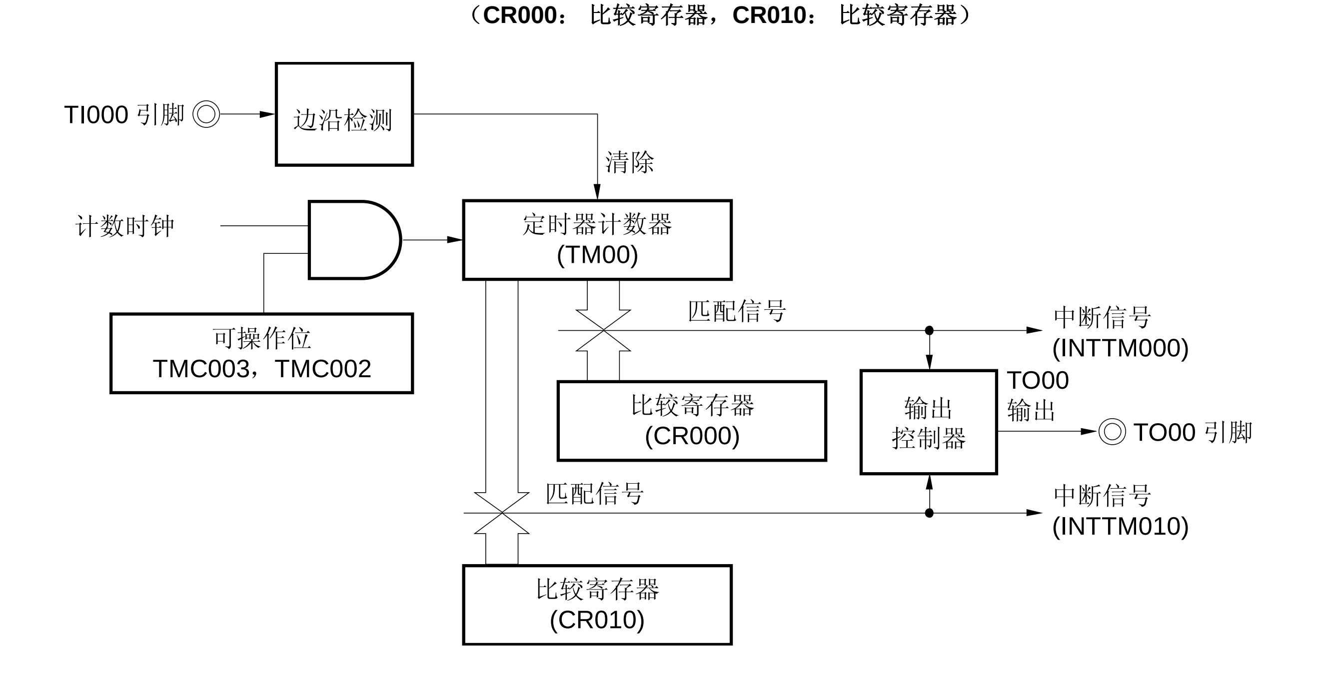
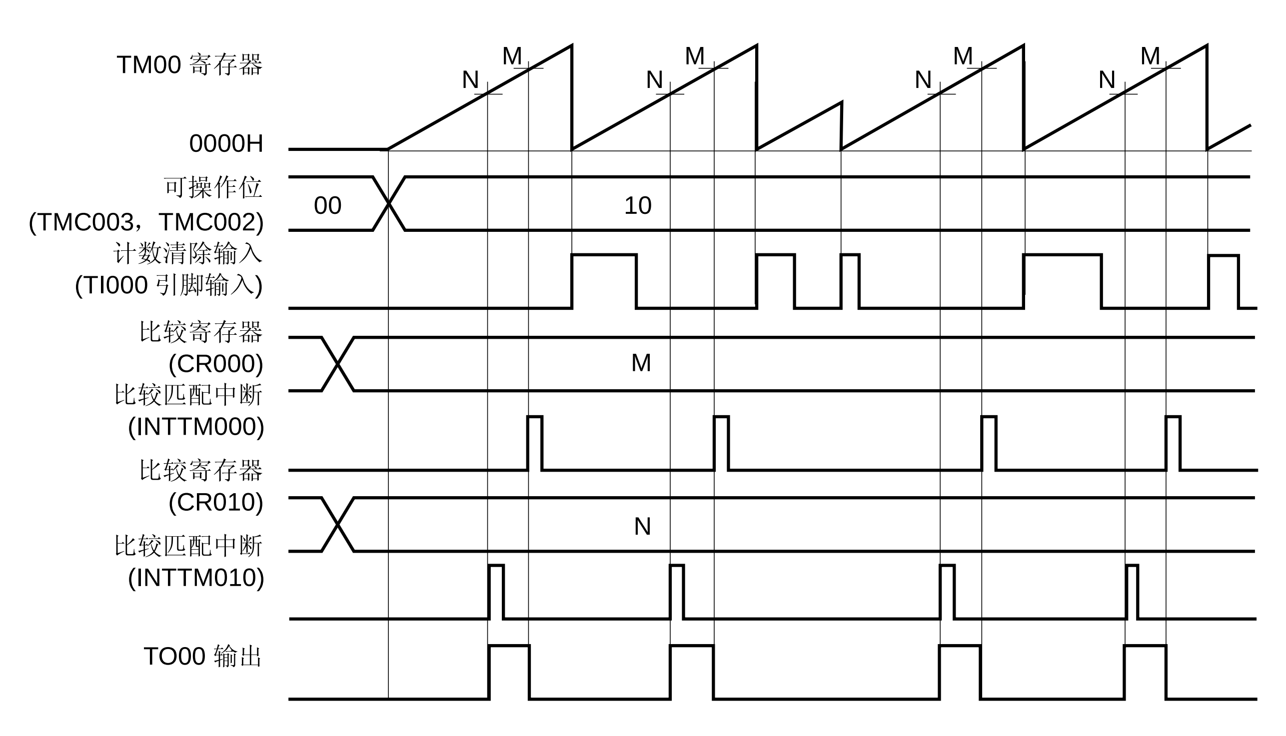
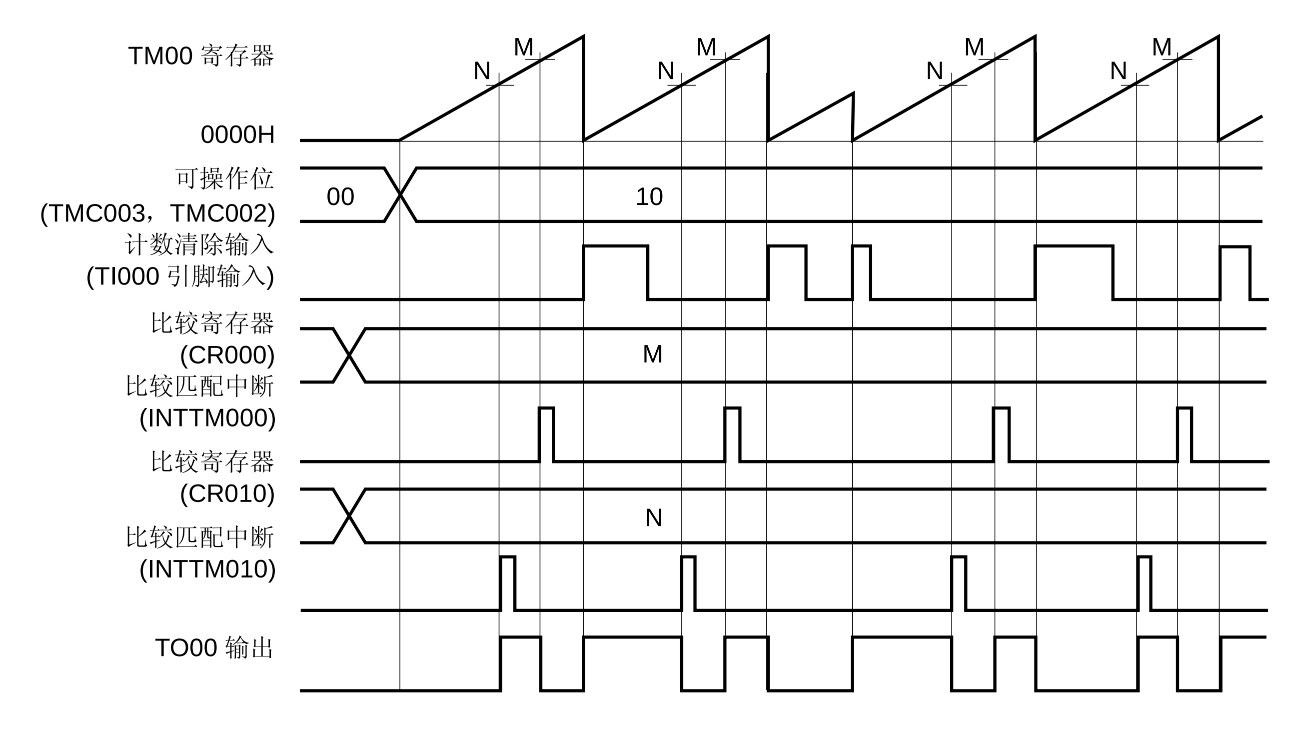


图 6-24. 通过输入 TI000 引脚的有效边沿进入清零&启动模式下的操作时序示例
（CR000： 比较寄存器，CR010： 比较寄存器）

（a）TOC00 = 13H，PRM00 = 10H，CRC00，= 00H，TMC00 = 08H



（b）TOC00 = 13H，PRM00 = 10H，CRC00，= 00H，TMC00 = 0AH



根据 16 位定时器模式控制寄存器 01（TMC00）第 1 位（TMC001）的设置，（a）和（b）会有如下变化。

（a） 当 TM00 与比较寄存器匹配时，反转 TO00 的输出电平。

（b） 当 TM00 与比较寄存器匹配或者检测到 TI000 引脚的有效边沿时，反转 TO00 的输出电平。

（2） 通过输入 TI000 引脚的有效边沿进入清零&启动模式的操作
（CR000： 比较寄存器，CR010： 捕获寄存器）

图 6-25. 通过输入 TI000 引脚的有效边沿进入清零&启动模式的框图
（CR000： 比较寄存器，CR010： 捕获寄存器）

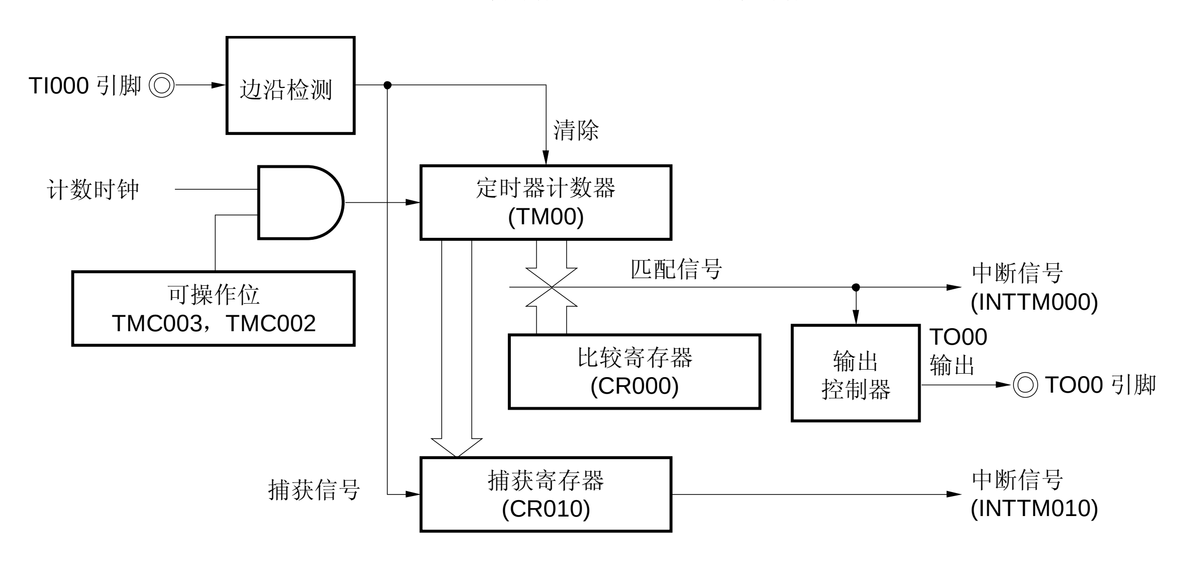
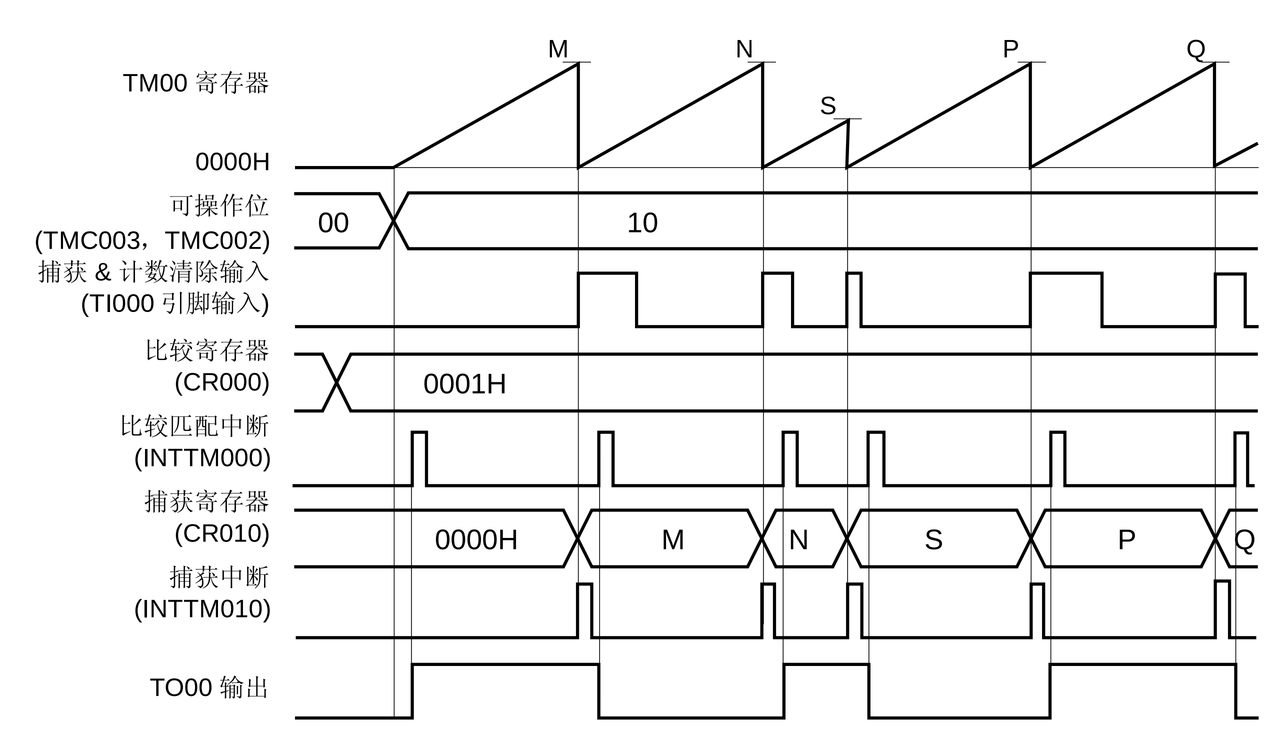


图 6-26. 通过输入 TI000 引脚的有效边沿进入清零&启动模式时序示例
（CR000： 比较寄存器，CR010： 捕获寄存器）（1/2）

（a）TOC00 = 13H，PRM00 = 10H，CRC00，= 04H，TMC00 = 08H，CR000 = 0001H

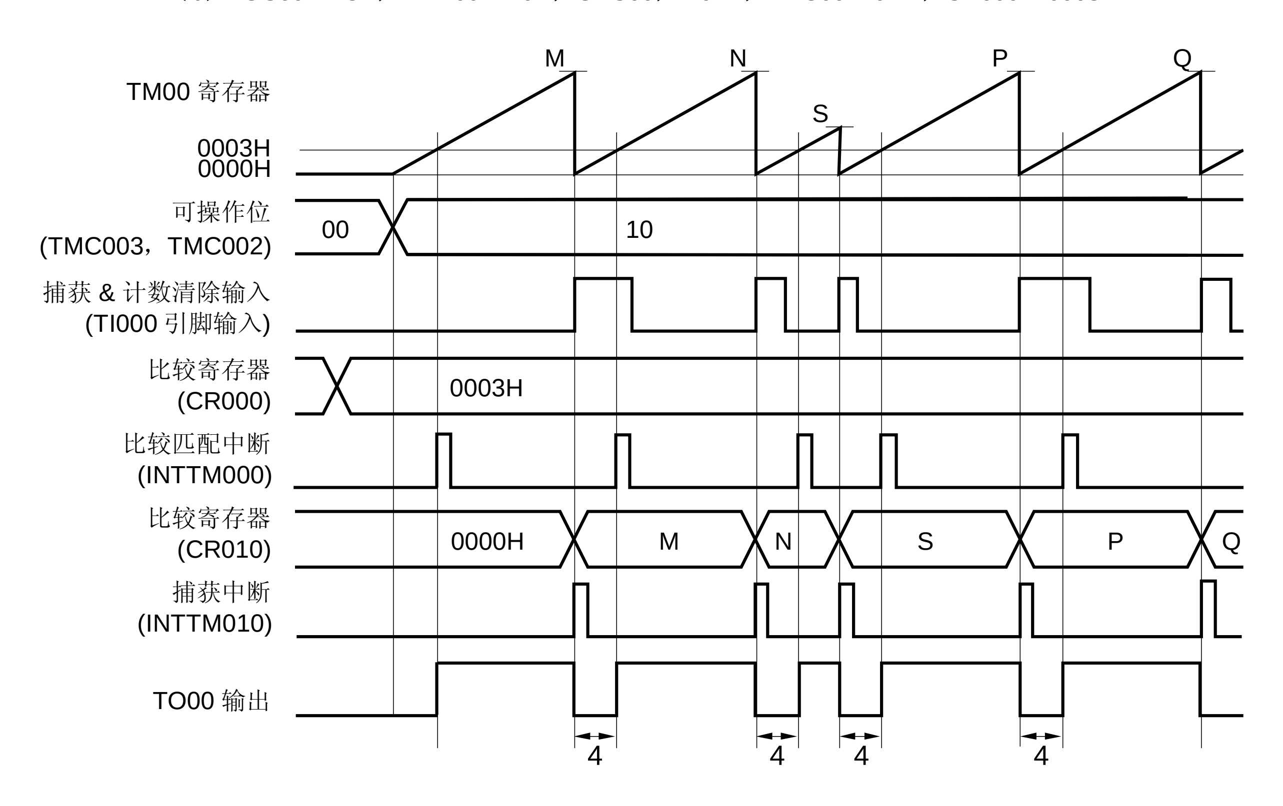


这是一个应用示例，当计数值被捕获&清除时，反转 TO00 的输出电平。

当检测到 TI000 引脚的有效边沿时，计数值捕获到 CR010 并将 TM00 清除（0000H）。当 TM00 的计数值为 0001H 时，产生比较匹配中断信号（INTTM000），并反转 TO00 的输出电平。

图 6-26. 通过输入 TI000 引脚的有效边沿进入清零&启动模式时序示例
（CR000： 比较寄存器，CR010： 捕获寄存器）（2/2）

（b）TOC00 = 13H，PRM00 = 10H，CRC00，= 04H，TMC00 = 0AH，CR000 = 0003H



这是一个应用示例，当计数值被捕获&清除时，从 TO00 引脚输出 CR000 的设置宽度（本例中为 4 个时钟）。

当检测到 TI000 引脚的有效边沿时，计数值捕获到 CR010、产生捕获中断信号（INTTM010）、将 TM00 清除（0000H）并反转 TO00 的输出。当 TM00 的计数值为 0003H 时（已经计数 4 个时钟），产生比较匹配中断信号（INTTM000）并反转 TO00 的输出电平。

（3） 通过输入 **TI000** 引脚的有效边沿进入清零**&**启动模式的操作
（**CR000**： 捕获寄存器，**CR010**： 比较寄存器）

图 **6-27.** 通过输入 **TI000** 引脚的有效边沿进入清零**&**启动模式的框图
（**CR000**： 捕获寄存器，**CR010**： 比较寄存器）

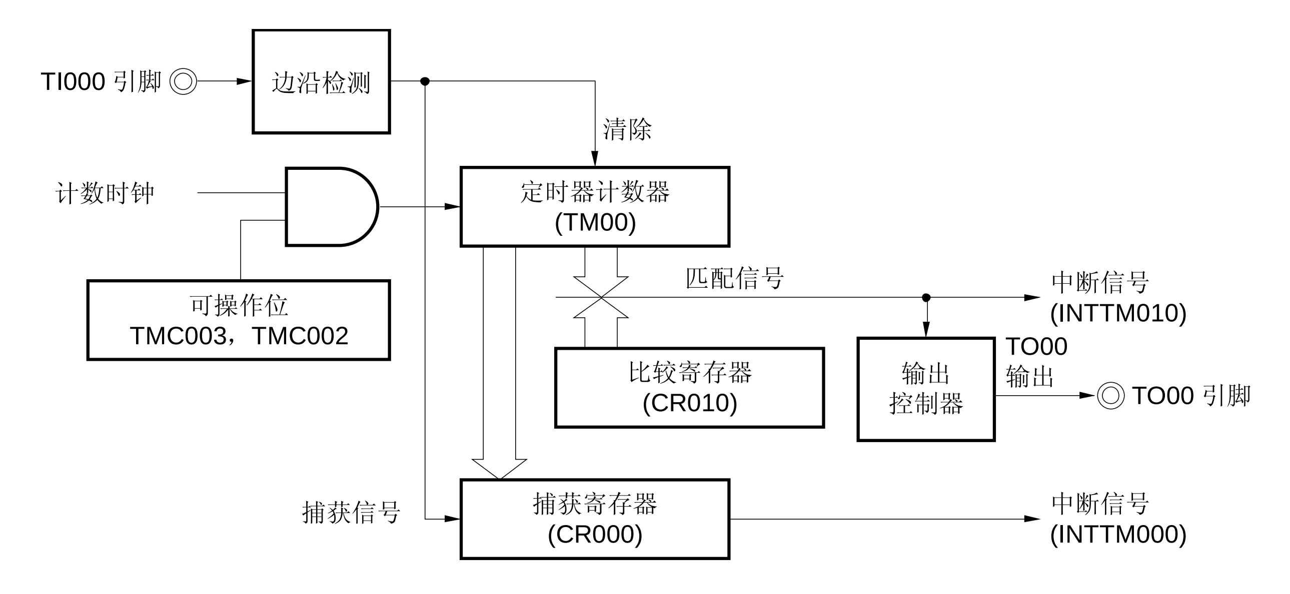
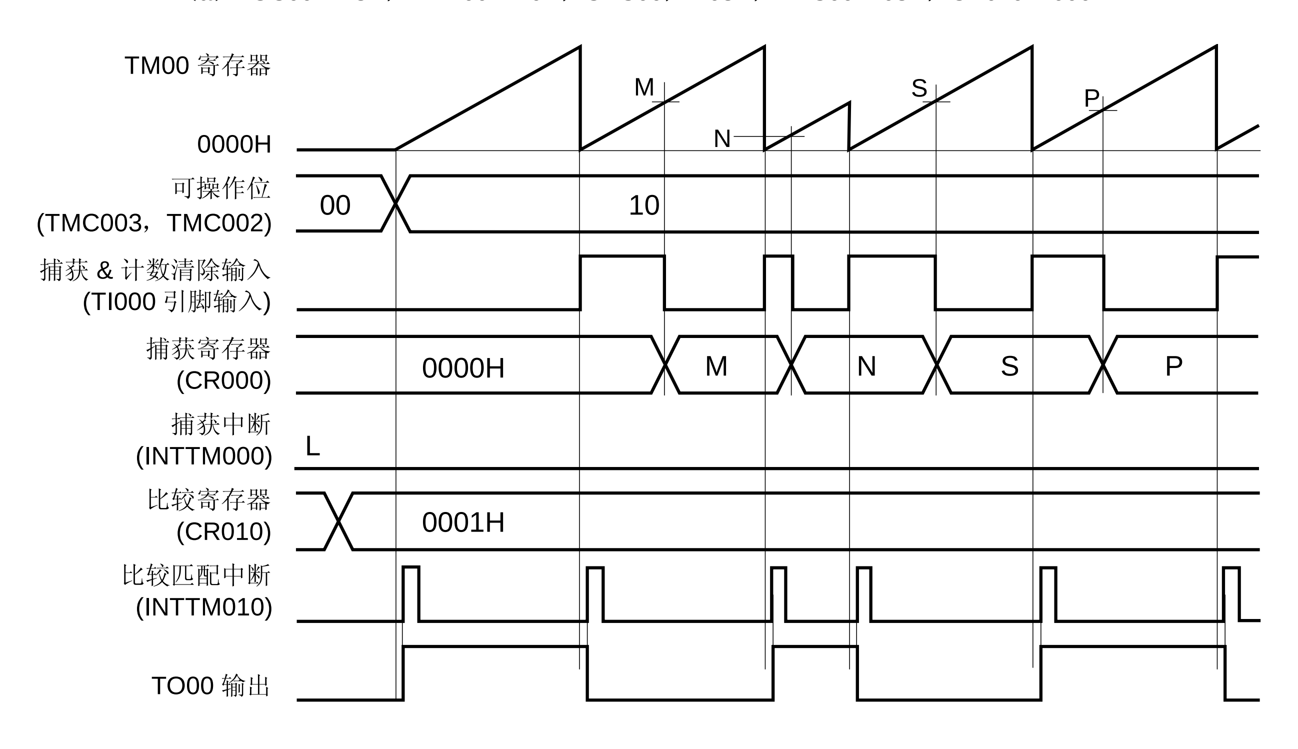


图 6-28. 通过输入 TI000 引脚的有效边沿进入清零&启动模式时序示例
（CR000： 捕获寄存器，CR010： 比较寄存器）（1/2）

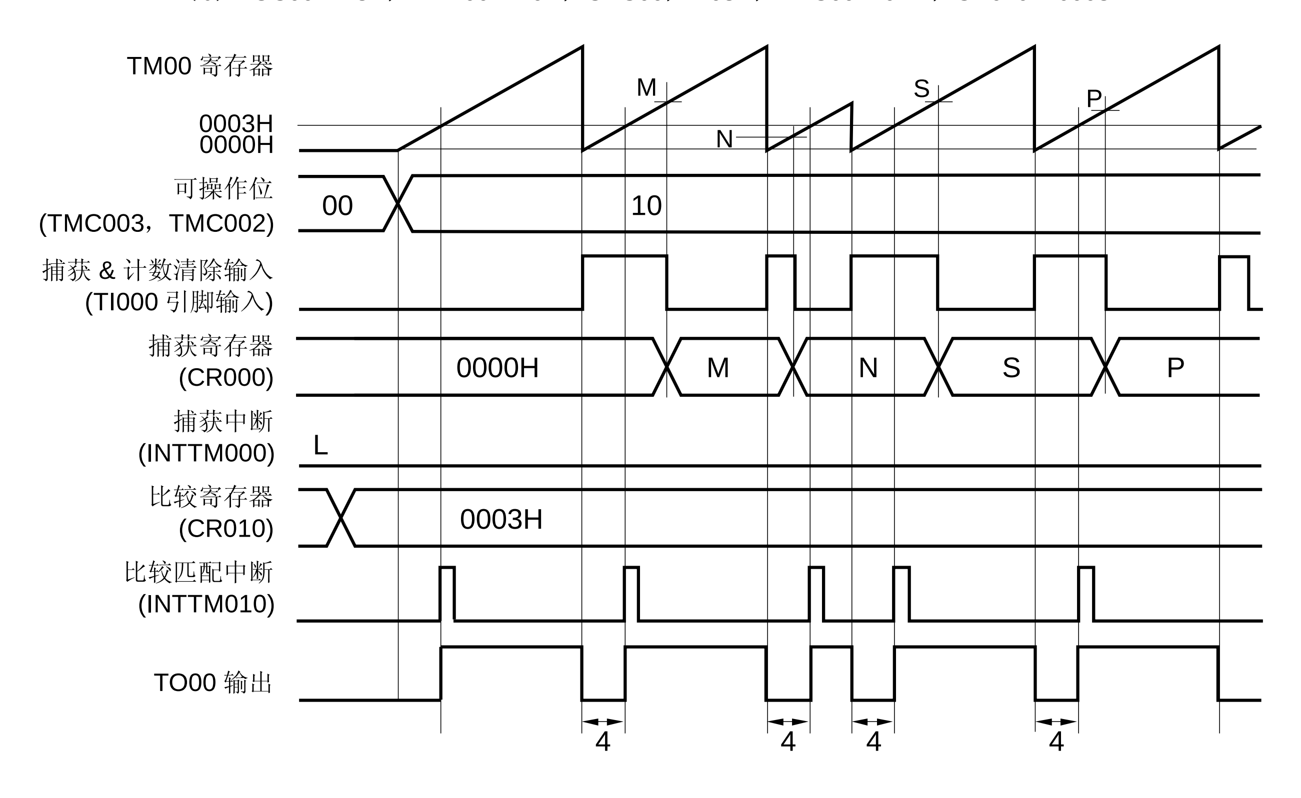
（a）TOC00 = 13H，PRM00 = 10H，CRC00，= 03H，TMC00 = 08H，CR010 = 0001H



这是一个应用示例，当计数值被捕获&清除时，反转 TO00 的输出电平。
在检测到 TI000 引脚的上升沿时，将 TM00 清除，而检测到 TI000 引脚的下降沿时，TM00 的计数值被捕获到 CR000。
当捕获/比较控制寄存器 00（CRC00）的第 1 位（CRC001）被设置为 1 时，在输入 TI000 引脚的信号反相处，TM00 的计数值被捕获到 CR000，但不产生捕获中断信号（INTTM000）。然而当检测到 TI010 引脚的有效边沿时，产生 INTTM000 信号。如果不使用 INTTM000 信号，则将其屏蔽。

图 6-28. 通过输入 TI000 引脚的有效边沿进入清零&启动模式时序示例
（CR000： 捕获寄存器，CR010： 比较寄存器）（2/2）

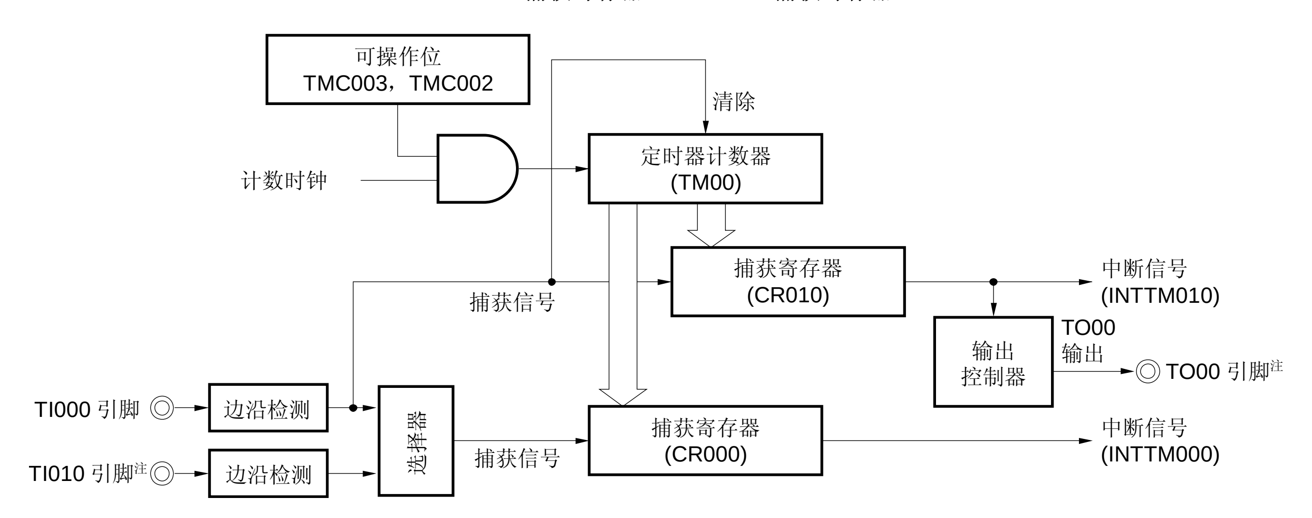
（b）TOC00 = 13H，PRM00 = 10H，CRC00，= 03H，TMC00 = 0AH，CR010 = 0003H



这是一个应用示例，当计数值被捕获&清除时，从 TO00 引脚输出 CR010 的设置宽度（本例中为 4 个时钟）。
在检测到 TI000 引脚的上升沿时，将 TM00 清除（为 0000H），而在检测到 TI000 引脚的下降沿时，TM00 的计数值被捕获到 CR000。当 TM00 被清零（为 0000H）时，TO00 的输出电平被反转，因为 TI00 引脚已经检测到上升沿，或者当 TM00 的值与比较寄存器（CR010）的值匹配时。
当捕获/比较控制寄存器 00（CRC00）的第 1 位（CRC001）为 1 时，在输入 TI000 引脚的信号反相处，TM00 的计数值被捕获到 CR000，但不产生捕获中断信号（INTTM000）。然而当检测到 TI010 引脚的有效边沿时，产生 INTTM000 中断信号。如果不使用 INTTM000 信号，则将其屏蔽。

（4） 通过输入 TI000 引脚的有效边沿进入清零&启动模式的操作
（CR000： 捕获寄存器，CR010： 捕获寄存器）

图 6-29. 通过输入 TI000 引脚的有效边沿进入清零&启动模式的框图
（CR000： 捕获寄存器，CR010： 捕获寄存器）

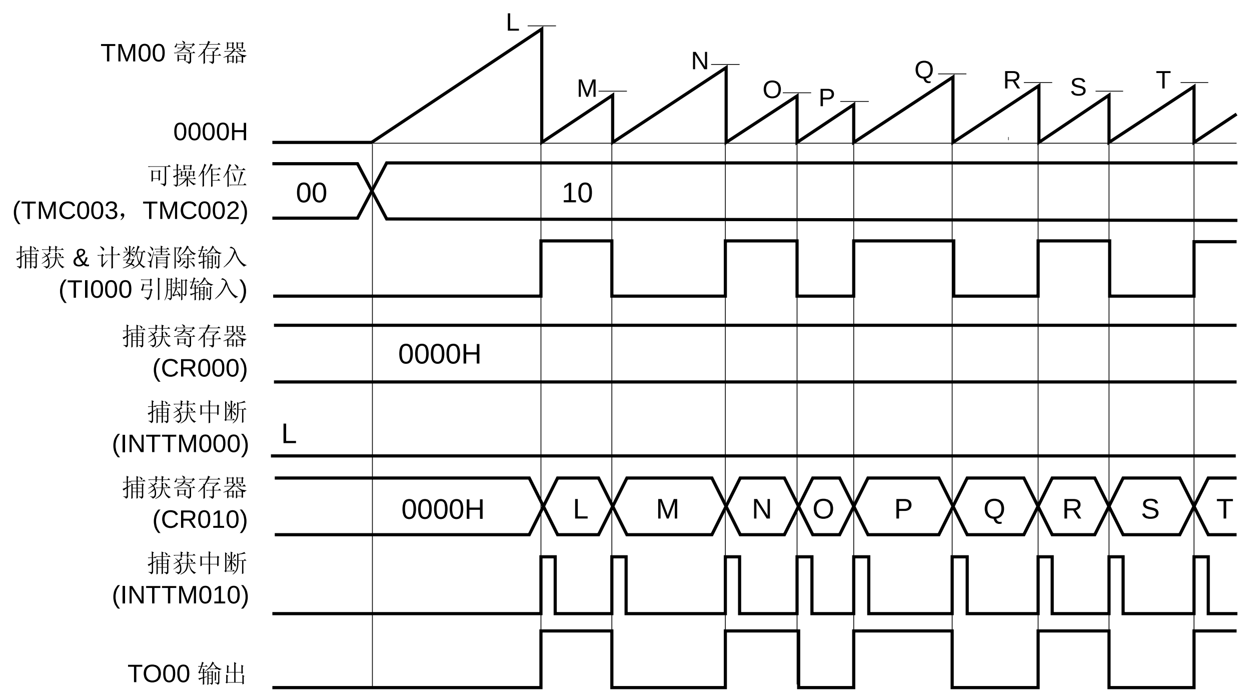


注　当用来检测 TI010 引脚的有效边沿时，不能用作定时器输出（TO00）。



图 6-30. 通过输入 TI000 引脚的有效边沿进入清零&启动模式时序示例
（CR000：捕获寄存器，CR010：捕获寄存器）（1/3）

（a）TOC00 = 13H，PRM00 = 30H，CRC00 = 05H，TMC00 = 0AH

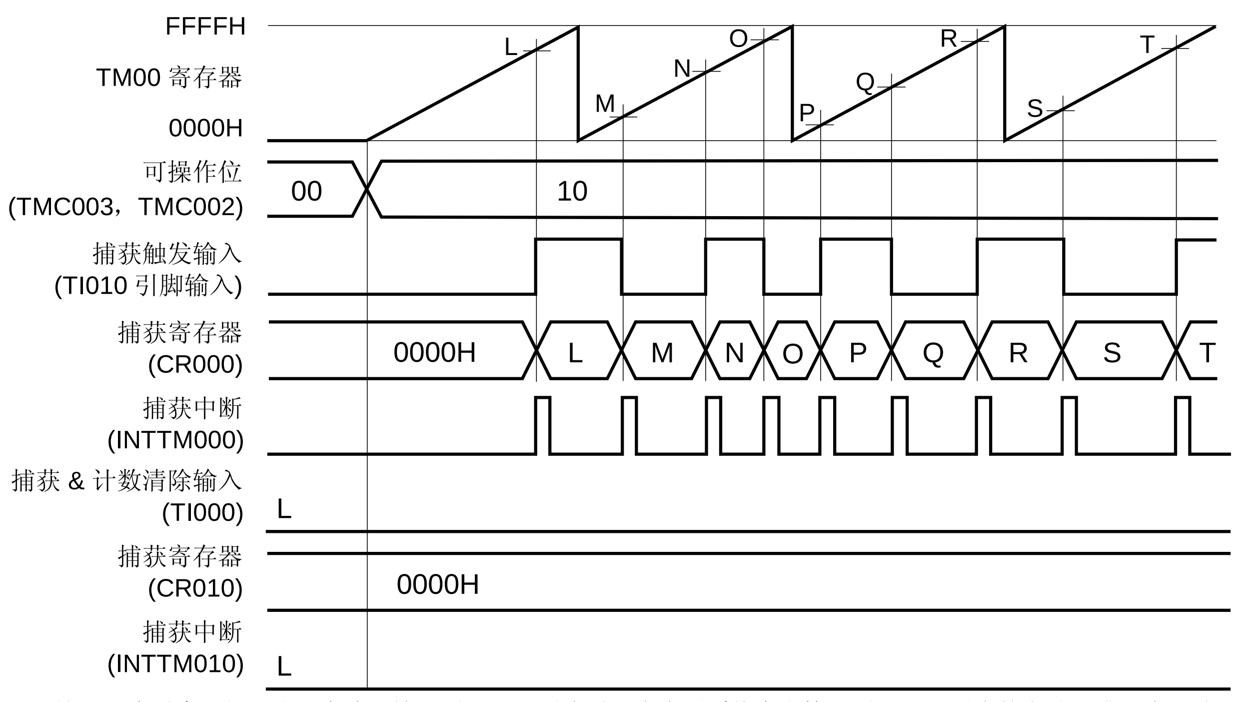


这是一个应用示例，当检测到 TI000 引脚的上升沿或下降沿时，计数值捕获到 CR010、清除 TM00 并反转 TO00 的输出。

在检测到 TI010 引脚的脉冲沿时，产生中断信号（INTTM000）。如果不使用 INTTM000 信号，则将其屏蔽。

图 6-30. 通过输入 **TI000** 引脚的有效边沿进入清零&启动模式时序示例
（**CR000**： 捕获寄存器，**CR010**： 捕获寄存器）（**2/3**）

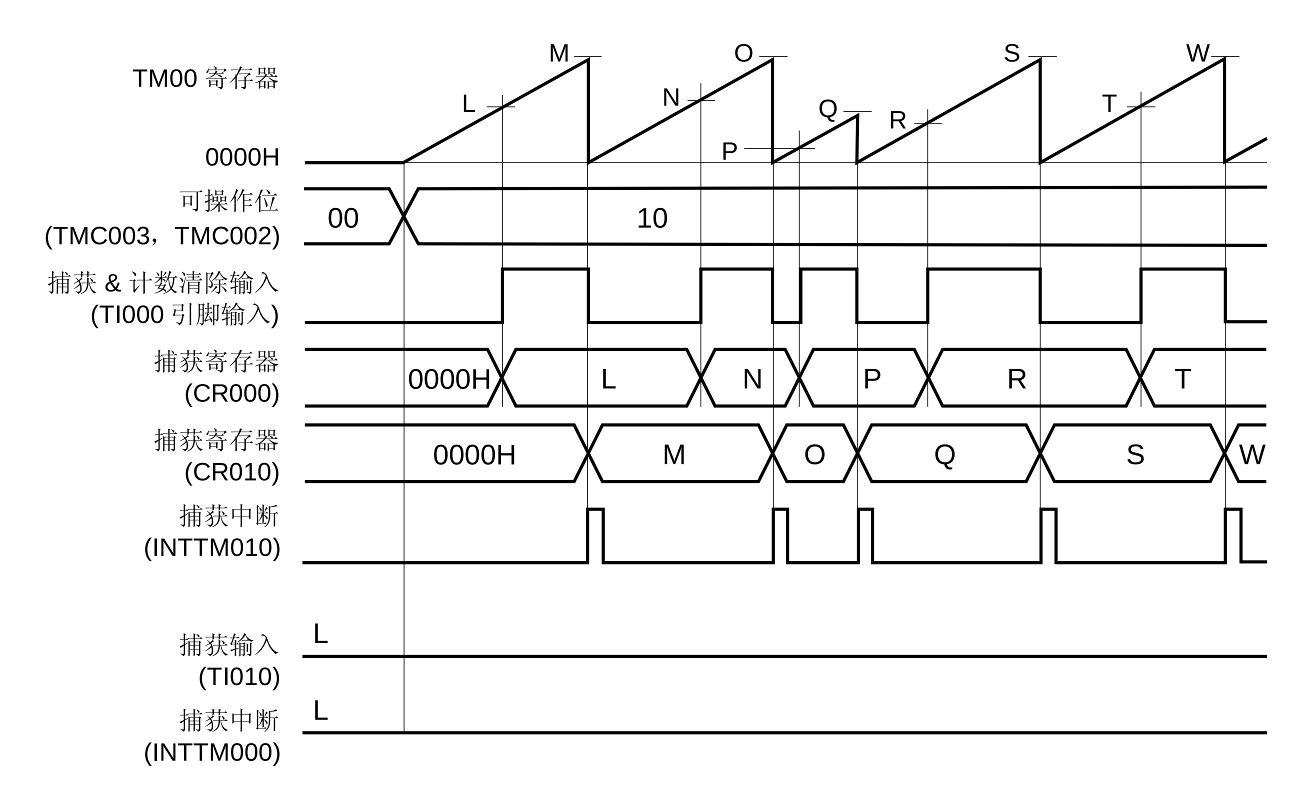
（**b**）**TOC00 = 13H**，**PRM00 = C0H**，**CRC00 = 05H**，**TMC00 = 0AH**



这是一个时序示例，当没有边沿输入到 TI000 引脚时，在应用系统中当检测到 TI010 引脚的上升沿或下降沿时，捕获到的计数值存入 CR000。

图 6-30. 通过输入 TI000 引脚的有效边沿进入清零&启动模式时序示例
（CR000： 捕获寄存器，CR010： 捕获寄存器）（3/3）

（c）TOC00 = 13H，PRM00 = 00H，CRC00 = 07H，TMC00 = 0AH



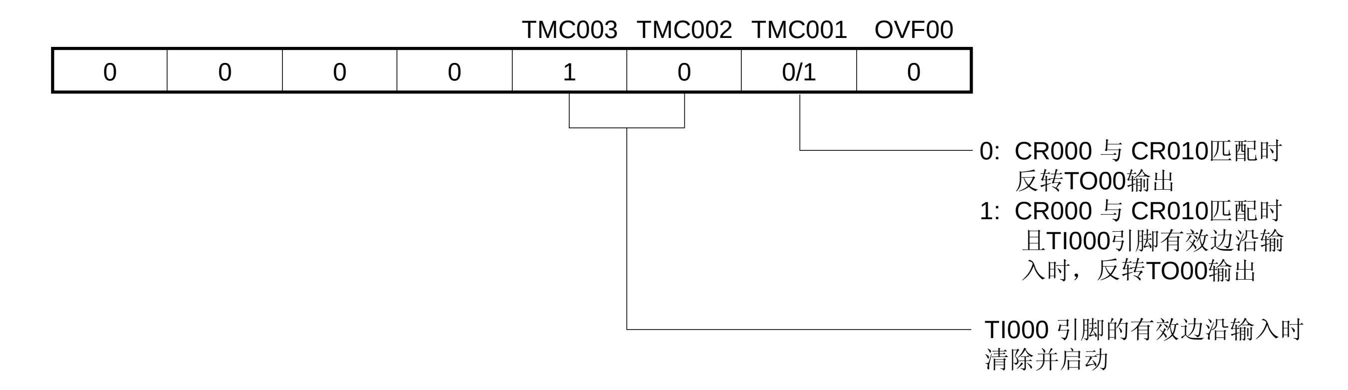
这是一个应用示例，对输入到 TI000 引脚的信号脉冲宽度进行测量。
通过设置 CRC00，在 TI000 引脚下降沿（也可以是上升沿）的反相时可以将计数值捕获到 CR000，而在 TI000 引脚下降沿将计数值捕获到 CR010。
可以通过下列表达式计算输入脉冲的高电平和低电平宽度。

- 高电平宽度 = [CR010 的值] – [CR000 的值] × [计数时钟周期]
- 低电平宽度 = [CR000 的值] × [计数时钟周期]

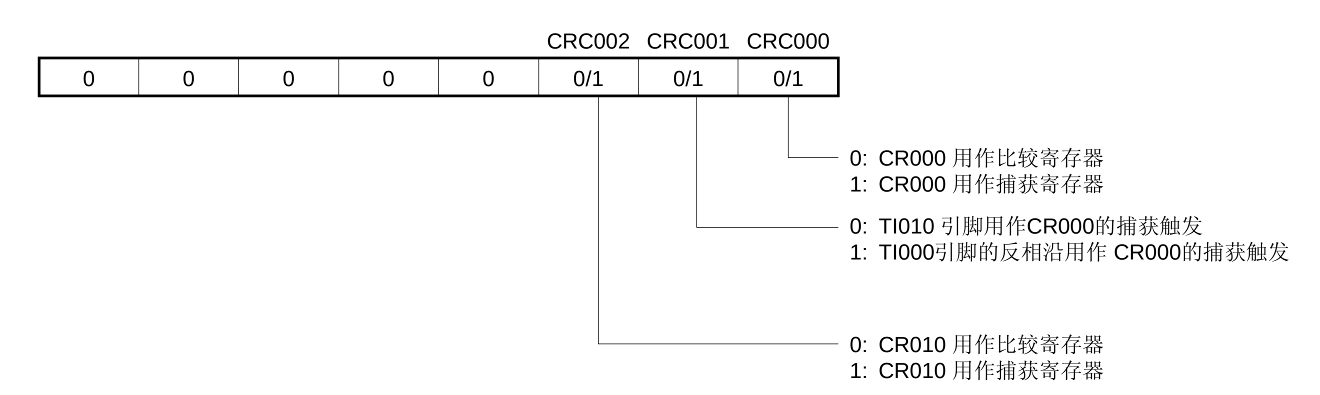
如果 TI000 引脚的反相沿被选择作为触发，将计数值捕获到 CR000，则不会产生 INTTM000 信号。在产生 INTTM010 信号后立即读取 CR000 和 CR010，可以测量脉冲宽度。
但是如果有预分频器模式寄存器 00（PRM00）的第 6 位和第 5 位（ES101 和 ES100）指定的有效边沿输入 TI010 引脚，则不会捕获计数值，但产生 INTTM000 信号。如果要测量 TI000 引脚的脉冲宽度，在不使用 INTTM000 信号时将其屏蔽。

图 6-31. 通过输入 TI000 引脚的有效边沿进入在清零&启动模式的寄存器设置示例（1/2）

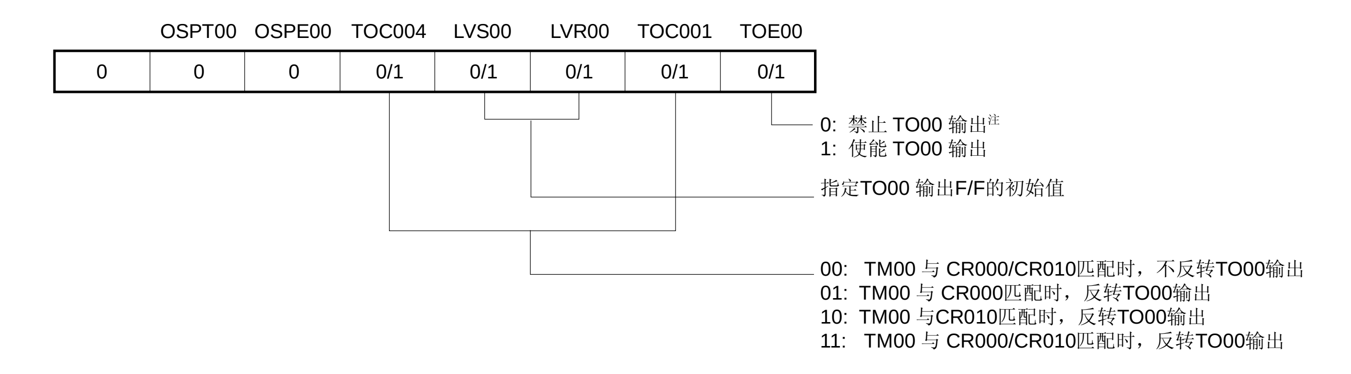
（a） 16 位定时器模式控制寄存器 00（TMC00）



（b） 捕获/比较控制寄存器 00（CRC00）



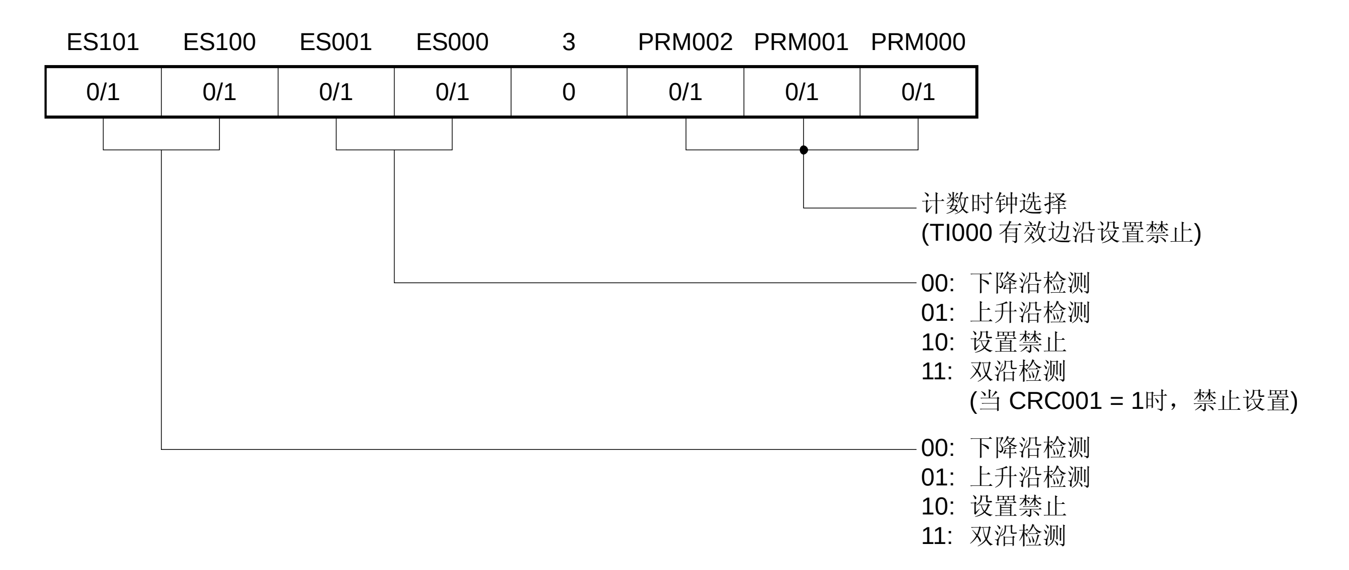
（c） 16 位定时器输出控制寄存器 00（TOC00）



注　当用来检测 TI010 引脚的有效边沿时，不能用作定时器输出（TO00）。

图 **6-31.** 通过输入 **TI000** 引脚的有效边沿进入在清零&启动模式的寄存器设置示例（**2/2**）

（**d**） 预分频模式寄存器 **00**（**PRM00**）



（**e**） **16** 位定时器计数器 **00**（**TM00**）

通过读取 TM00，获得计数值。

（**f**） **16** 位捕获**/**比较寄存器 **000**（**CR000**）

当该寄存器用作比较寄存器并且其值与 TM00 的计数值匹配时，产生中断信号（INTTM000）。但 TM00 的计数值不清除。

如果要将该寄存器用作捕获寄存器，则应选择 TI000 或 TI010 引脚[注]输入作为捕获触发。当检测到捕获触发的有效边沿时，将 TM00 的计数值存入 CR000。

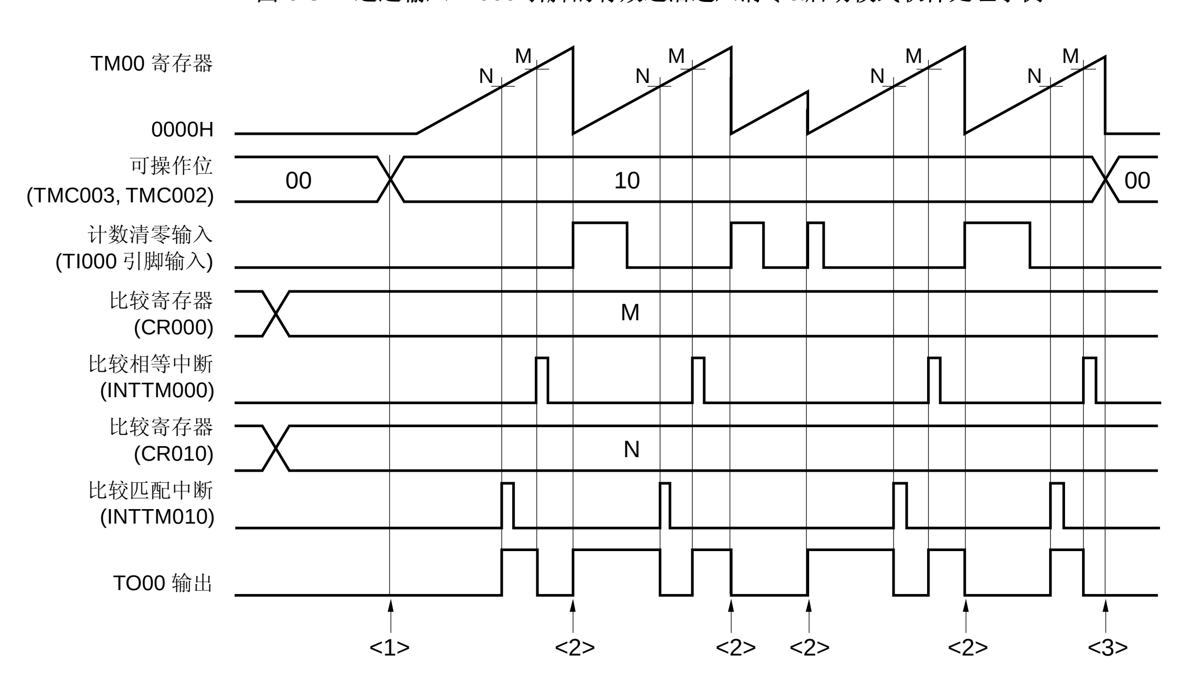
注 当用来检测 TI010 引脚的有效边沿时，不能用作定时器输出（TO00）。

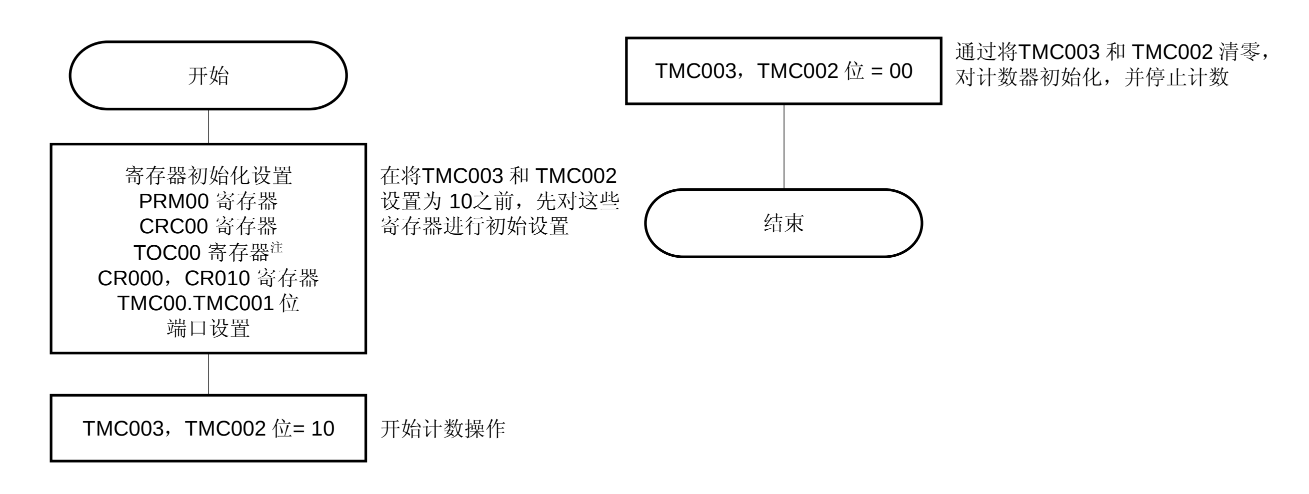
（**g**） **16** 位捕获**/**比较寄存器 **010**（**CR010**）

当该寄存器用作比较寄存器并且其值与 TM00 的计数值匹配时，产生中断信号（INTTM010）。但 TM00 的计数值不清除。

当该寄存器用作捕获寄存器，则 TI000 引脚输入作为捕获触发。当检测到捕获触发的有效边沿时，将 TM00 的计数值存入 CR010。

图 6-32. 通过输入 TI000 引脚的有效边沿进入清零&启动模式软件处理示例







注 设置 TOC00 时必须特别注意。详情参见 **6.3（3） 16 位定时器输出控制寄存器 00（TOC00）**。

6.4.5 自由运行定时器操作

当 16 位定时器模式控制寄存器 00（TMC00）的第 3 位和第 2 位（TMC003 与 TMC002）为 01（自由运行定时器模式）时，16 位定时器/事件计数器 00 随着计数时钟同步向上继续计数。当计数值上升到 FFFFH 时，在下一个时钟将溢出标志（OVF00）设置为 1、将 TM00 清除（为 0000H）并继续计数。通过软件执行 CLR 指令将 OVF00 清除为 0。

有以下三种类型的自由运行定时器操作可供使用。

- CR000 和 CR010 都用作比较寄存器。
- CR000 与 CR010 一个用作比较寄存器，另一个用作捕获寄存器。
- CR000 和 CR010 都用作捕获寄存器。

备注 **1.** 如需了解 I/O 引脚的设置，参见 **6.3（6） 端口模式寄存器 3（PM3）**。
2. 要了解如何使能 INTTM000 信号中断，参见**第二十章 中断功能**。

（1） 自由运行定时器模式操作
（CR000： 比较寄存器，CR010： 比较寄存器）

图 **6-33.** 自由运行定时器模式的框图
（CR000： 比较寄存器，CR010： 比较寄存器）

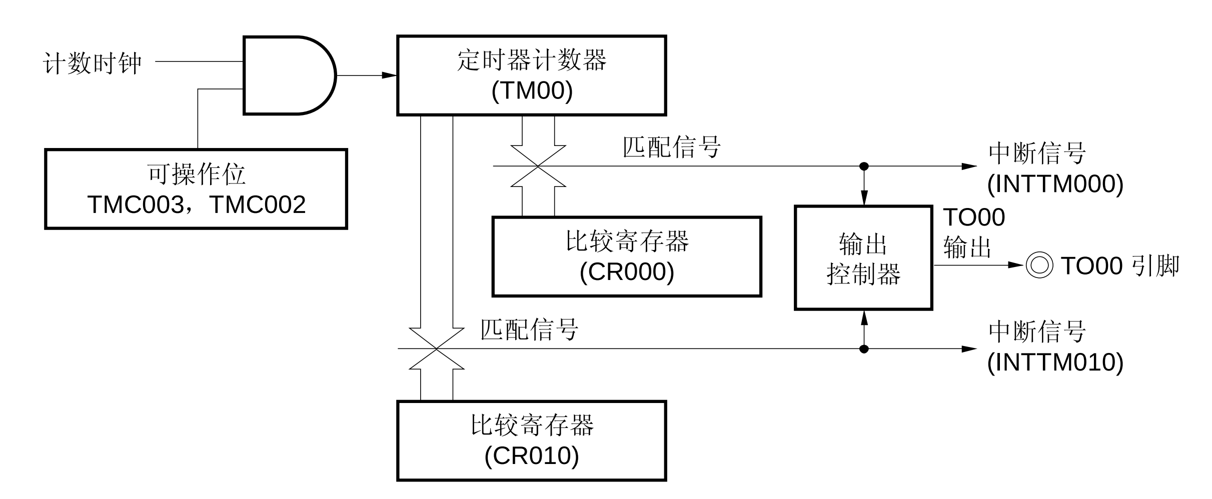
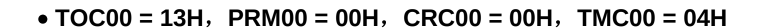
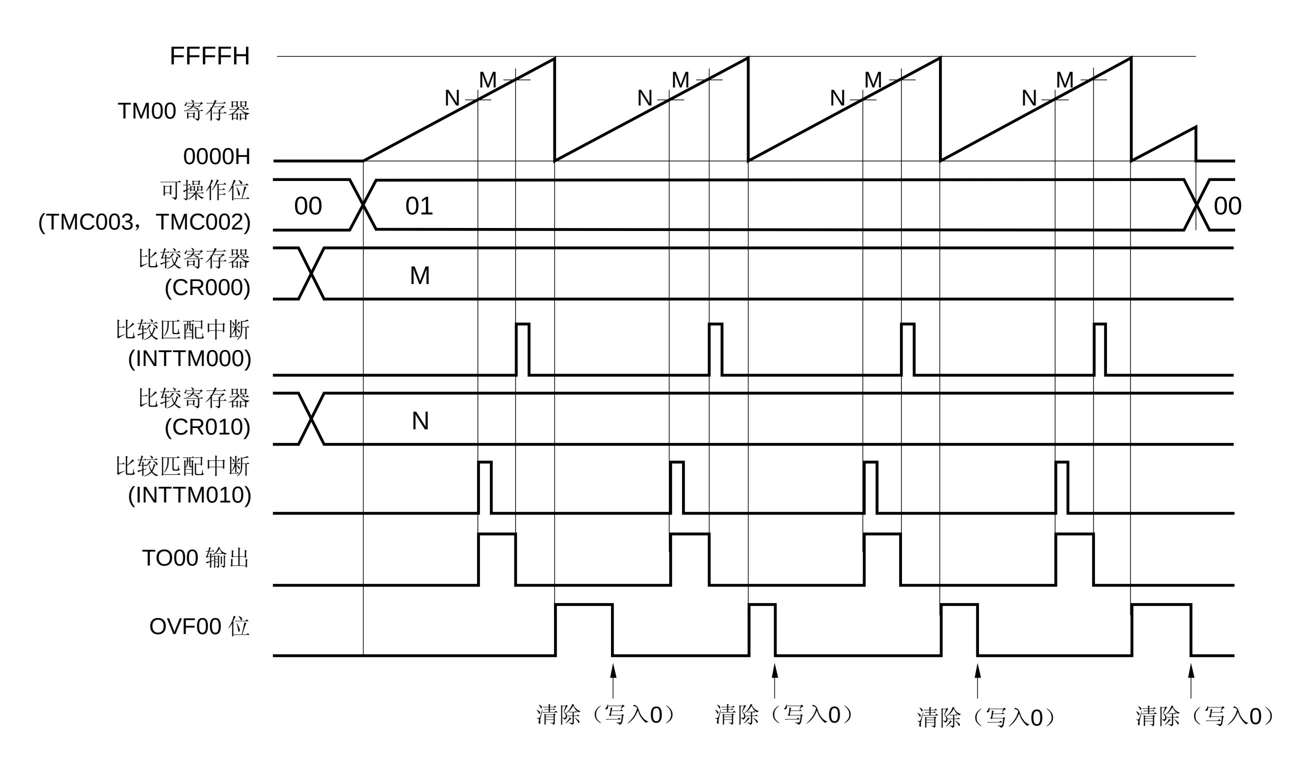


图 **6-34.** 自由运行定时器模式的时序示例
（**CR000**： 比较寄存器，**CR010**： 比较寄存器）


• **TOC00 = 13H，PRM00 = 00H，CRC00 = 00H，TMC00 = 04H**



这是一个应用示例，两个比较寄存器都用作自由运行定时器模式。
每当 TM00 的计数值与 CR000 或 CR010 的设置值匹配时，反转 TO00 的输出电平。当计数值与寄存器值匹配时，产生 INTTM000 或 INTTM010 信号。

（**2**） 自由运行定时器模式操作
（**CR000**： 比较寄存器，**CR010**： 捕获寄存器）

图 **6-35.** 自由运行定时器模式的框图
（**CR000**： 比较寄存器，**CR010**： 捕获寄存器）

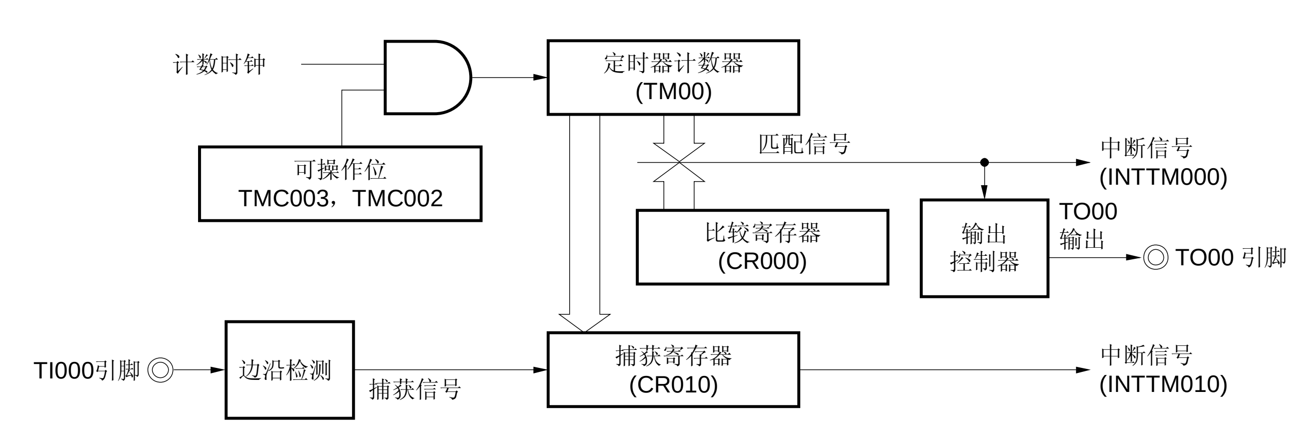
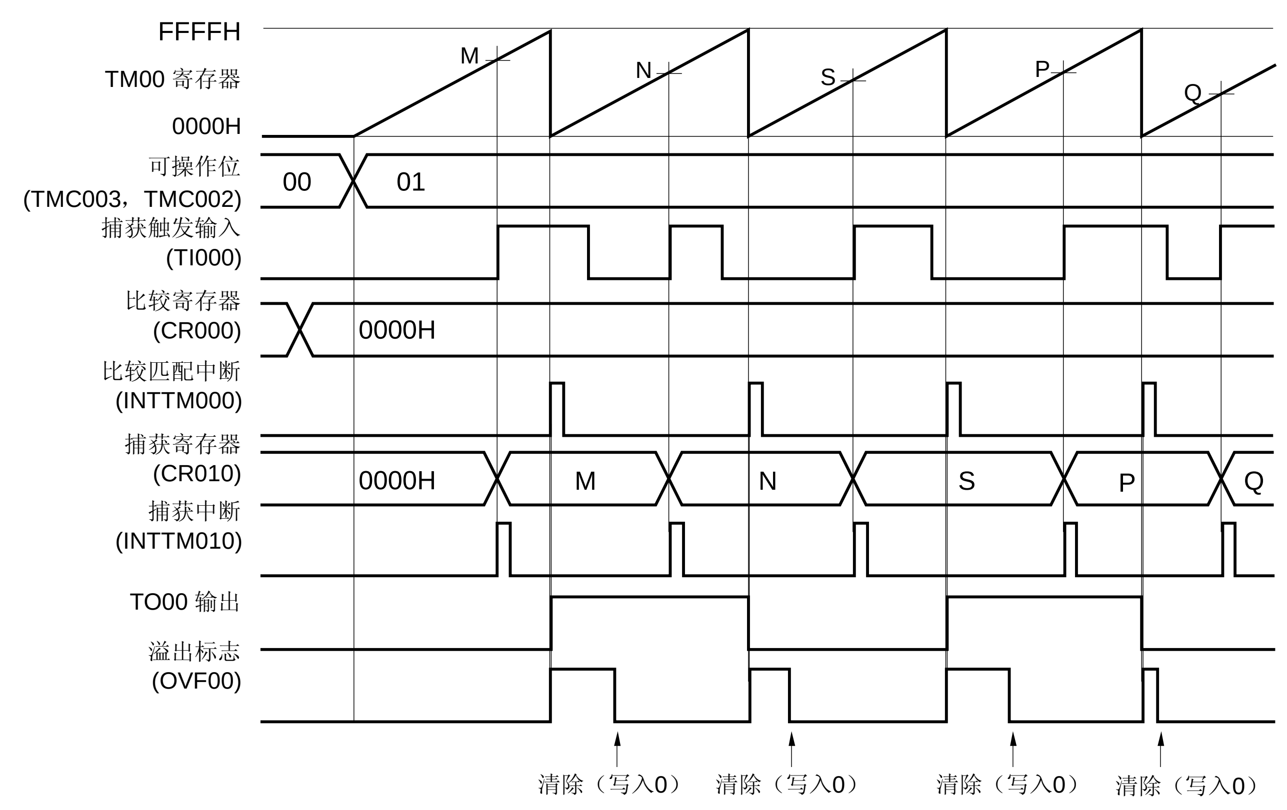


图 6-36. 自由运行定时器模式时序示例
（CR000： 比较寄存器，CR010： 捕获寄存器）

• TOC00 = 13H，PRM00 = 10H，CRC00 = 04H，TMC00 = 04H

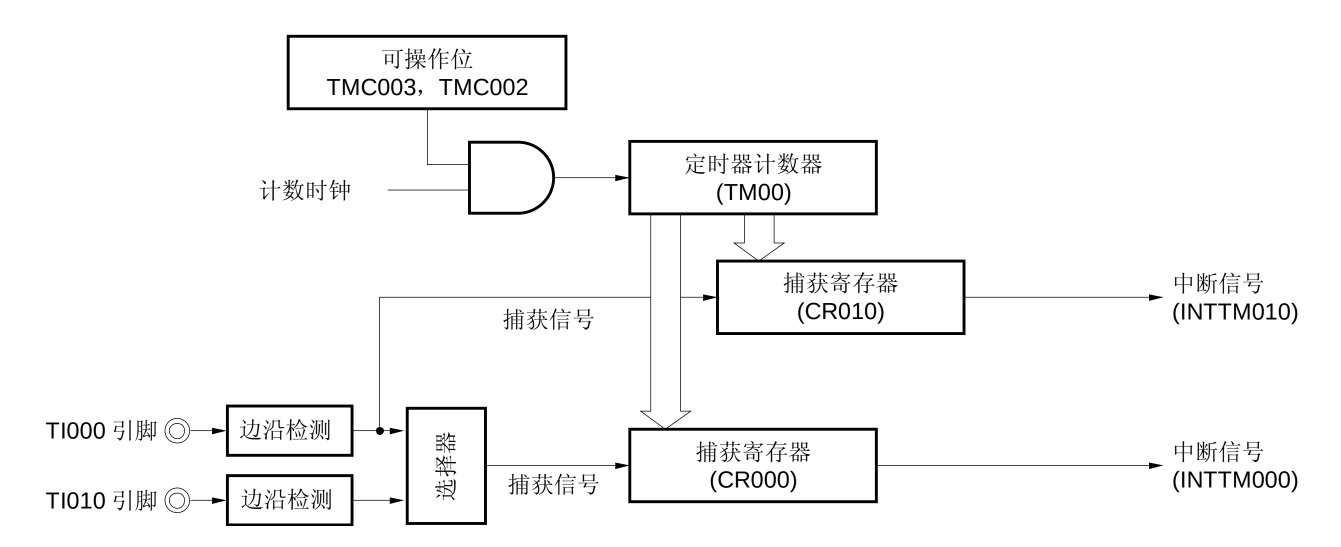


这是一个应用示例，在自由运行定时器模式下同时使用一个比较寄存器和一个捕获寄存器。

在本例中，每当 TM00 的计数值与 CR000（比较寄存器）的设置值匹配时，反转 TO00 的输出电平，并产生 INTTM000 信号。此外，每当检测到 TI000 引脚的有效边沿时，产生 INTTM010 信号，并将 TM00 的计数值捕获到 CR010。

（3） 自由运行定时器模式操作
（CR000： 捕获寄存器，CR010： 捕获寄存器）

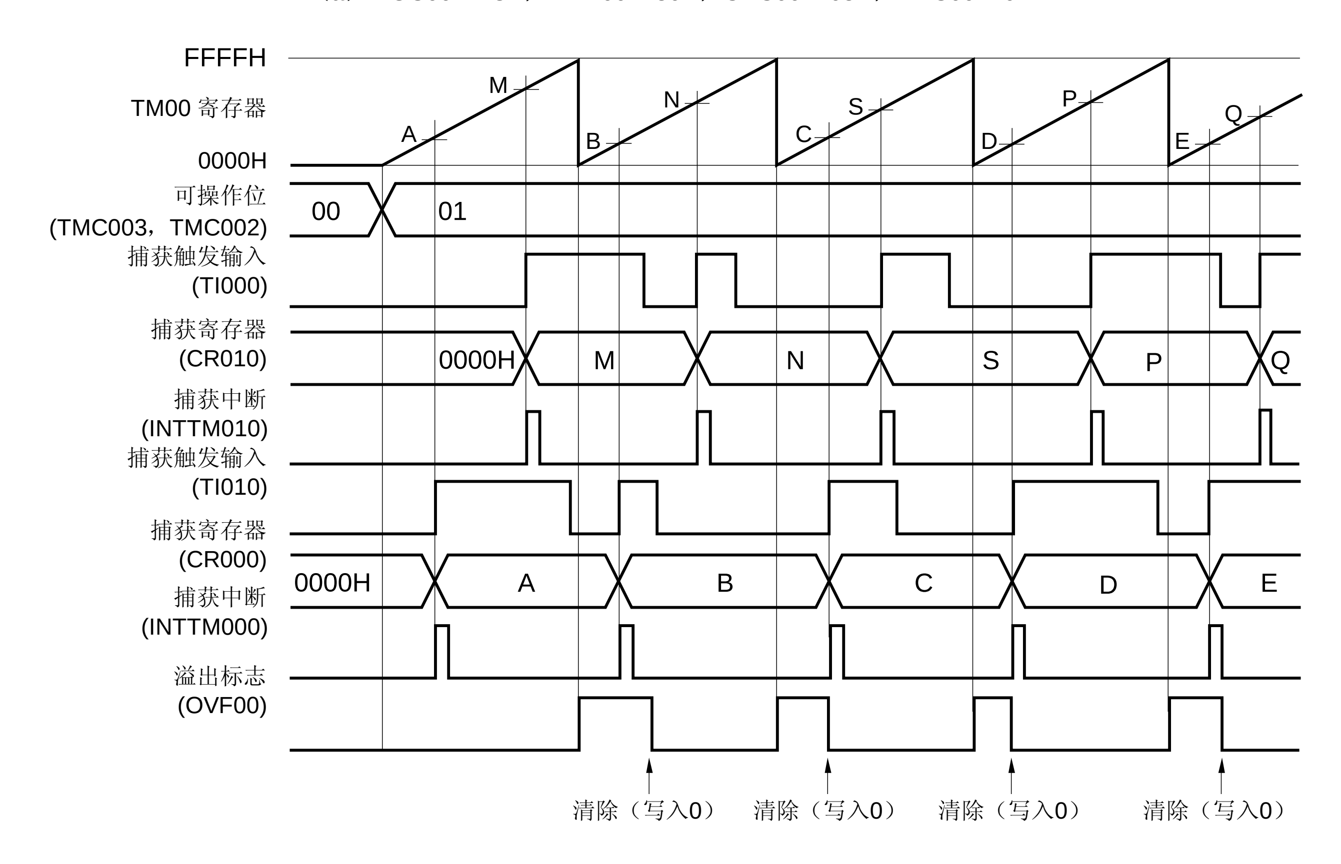
图 6-37. 自由运行定时器模式的框图
（CR000： 捕获寄存器，CR010： 捕获寄存器）



备注 在自由运行定时器模式下，如果 CR000 与 CR010 用作捕获寄存器，则不反转 TO00 的输出电平。
但如果 16 位定时器模式控制寄存器 00（TMC00）的第 1 位（TMC001）被设置为 1，则每次检测到 TI000 引脚的有效边沿时，可以反转。

图 **6-38.** 自由运行定时器模式的时序示例
（**CR000**： 捕获寄存器，**CR010**： 捕获寄存器）（**1/2**）

（**a**） **TOC00 = 13H，PRM00 = 50H，CRC00 = 05H，TMC00 = 04H**

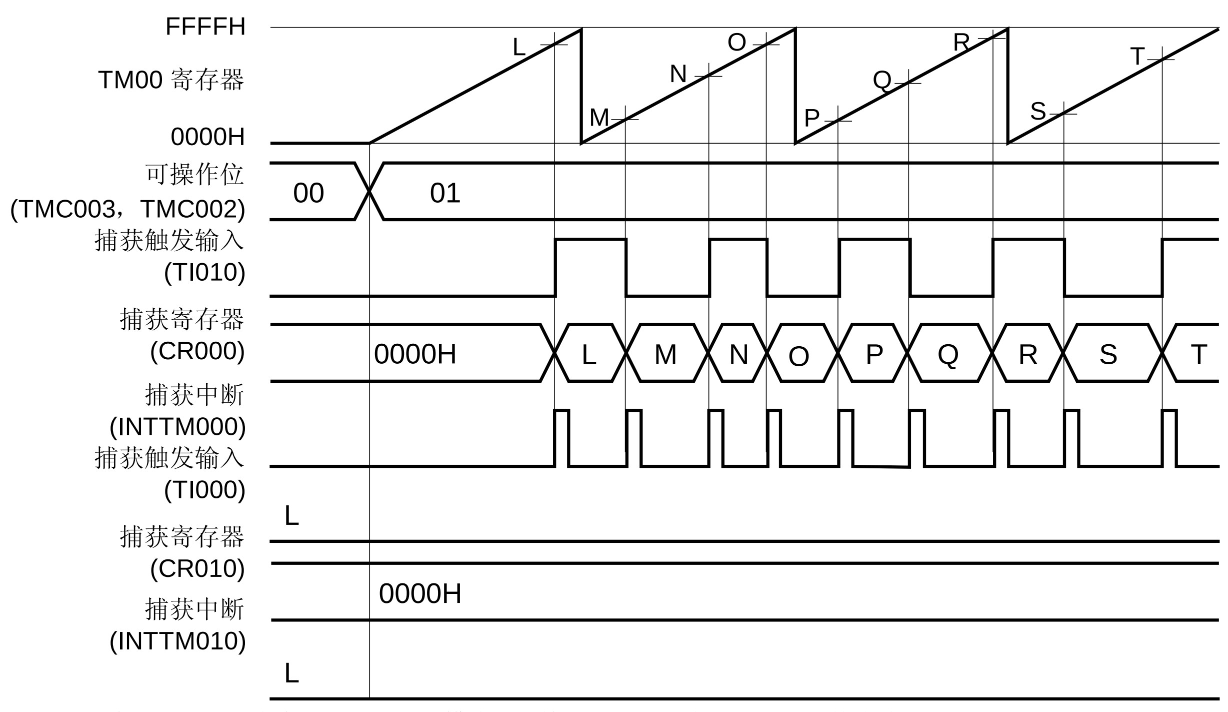


这是一个应用示例，自由运行定时器模式下，在各自触发信号的有效边沿捕获的计数值存入各自的捕获寄存器中。当检测到 TI000 引脚输入的有效边沿时，捕获计数值并存入 CR010，而当检测到 TI010 引脚输入的有效边沿时，捕获计数值并存入 CR000。

图 **6-38.** 自由运行定时器模式的时序示例

（**CR000**： 捕获寄存器，**CR010**： 捕获寄存器）（**2/2**）

（**b**） **TOC00 = 13H，PRM00 = C0H，CRC00 = 05H，TMC00 = 04H**

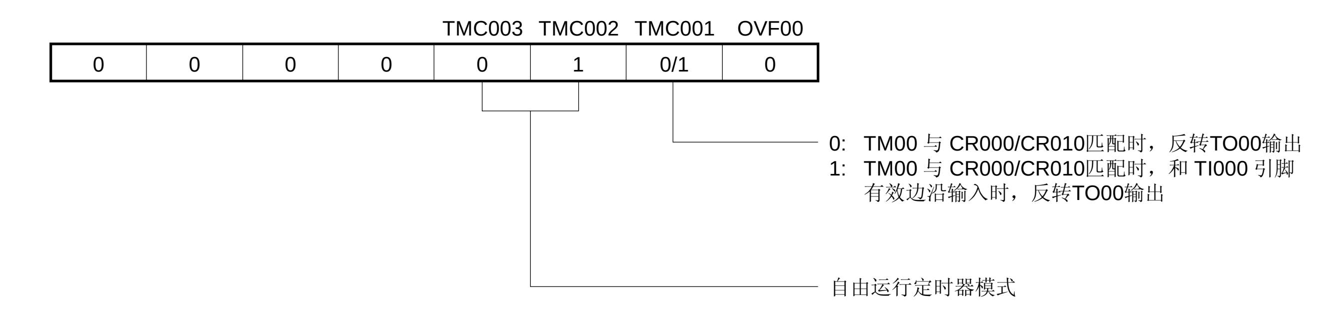


这是一个应用示例，在自由运行定时器模式下，检测 TI010 引脚的双沿，计数值被捕获到 CR000。

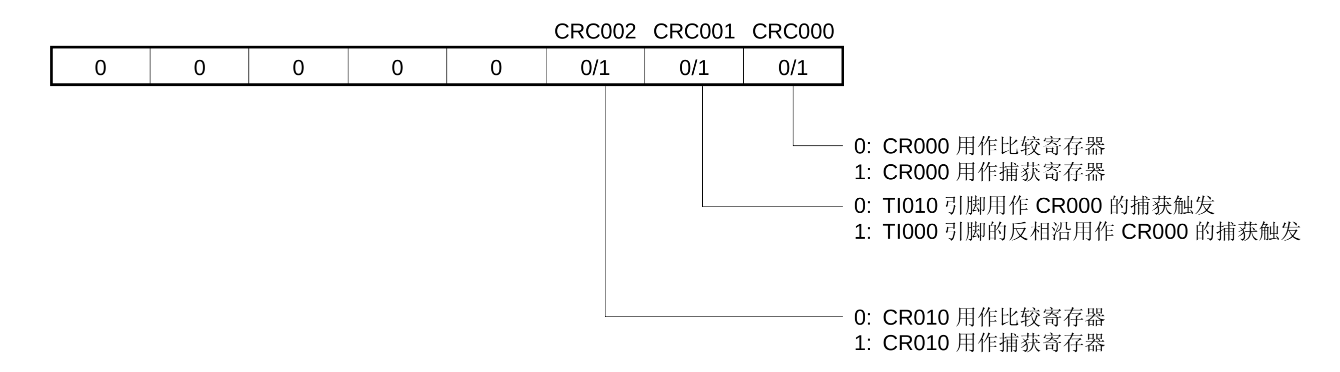
当 CR000 与 CR010 都用作捕获寄存器，并且仅检测到 TI010 引脚的有效边沿时，计数值不能被捕获到 CR010。

图 6-39. 自由运行定时器模式下寄存器设置示例（1/2）

（a） 16 位定时器模式控制寄存器 00（TMC00）



（b） 捕获/比较控制寄存器 00（CRC00）



（c） 16 位定时器输出控制寄存器 00（TOC00）

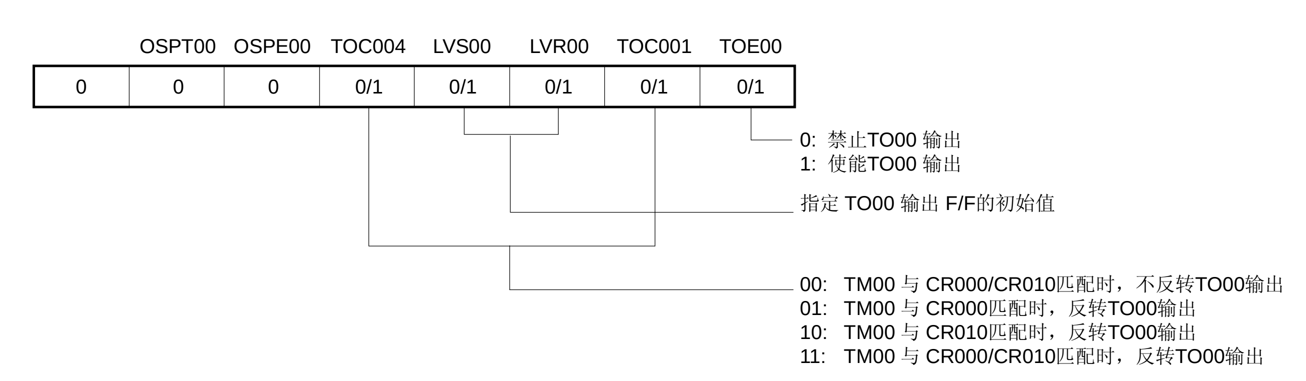
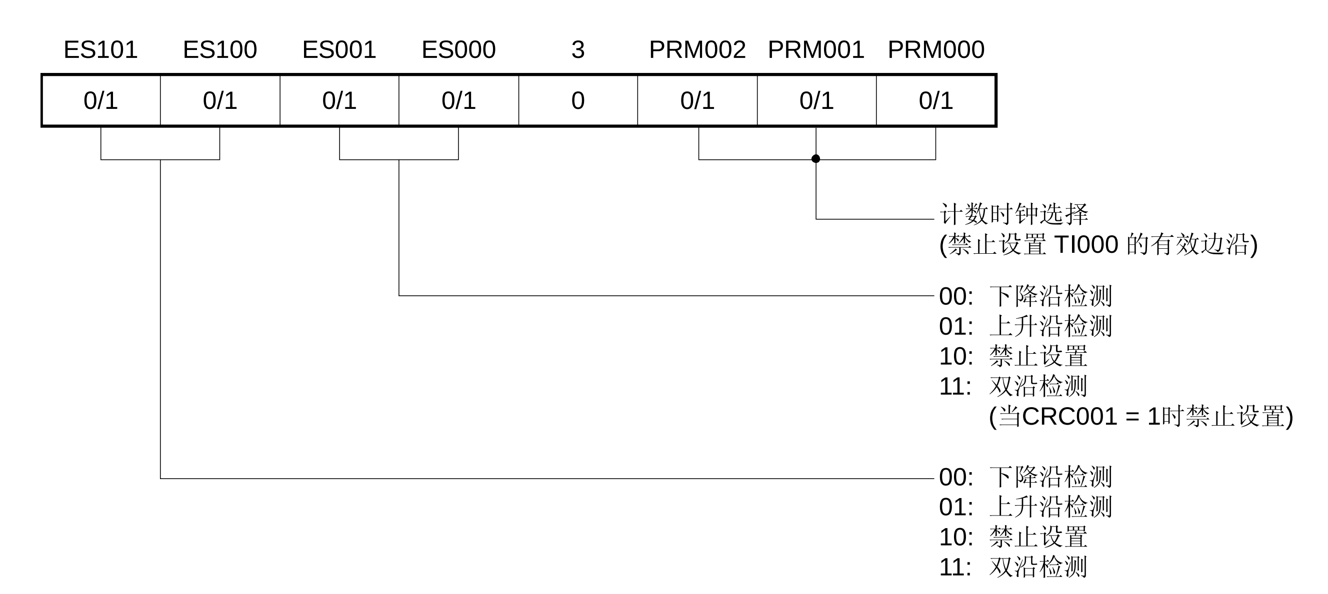


图 **6-39.** 自由运行定时器模式下寄存器设置示例（**2/2**）

（**d**） 预分频模式寄存器 **00**（**PRM00**）



（**e**） **16** 位定时器计数器 **00**（**TM00**）

通过读取 TM00，获得计数值。

（**f**） **16** 位捕获**/**比较寄存器 **000**（**CR000**）

当该寄存器用作比较寄存器并且其值与 TM00 的计数值匹配时，产生中断信号（INTTM000）。但 TM00 的计数值不清除。

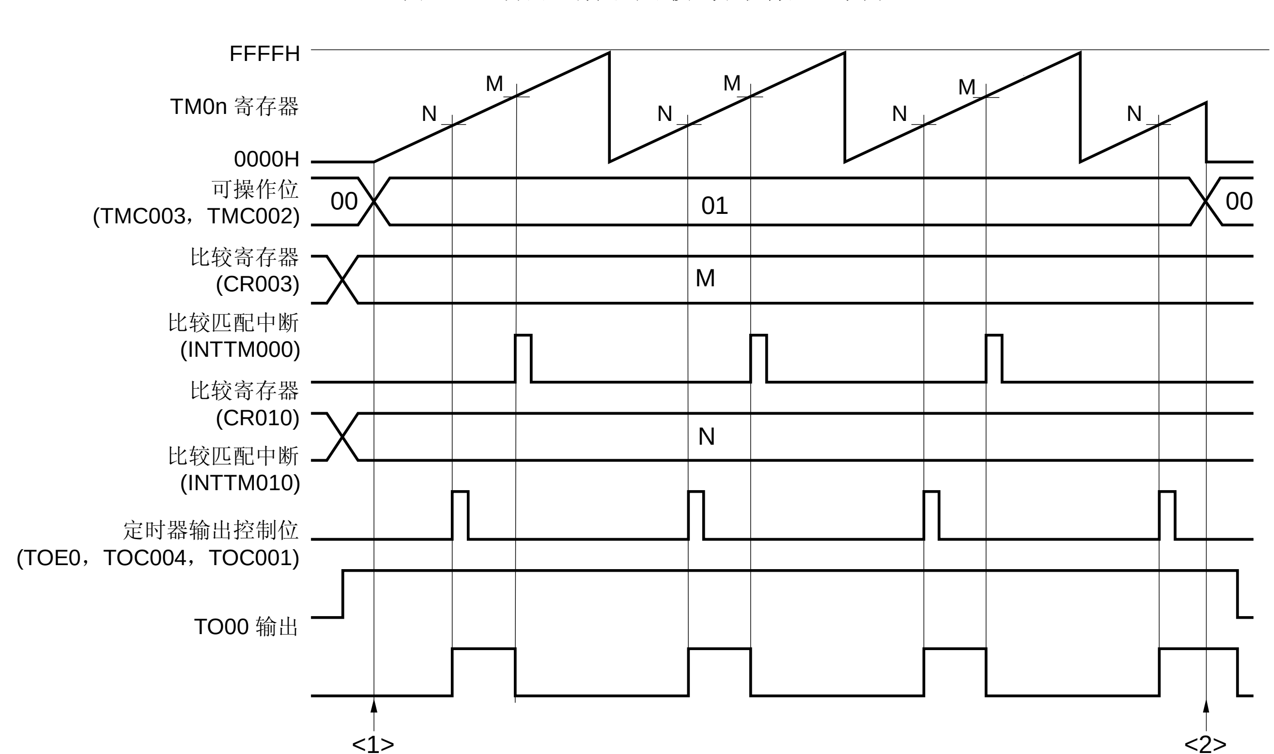
如果要将该寄存器用作捕获寄存器，则应选择 TI000 或 TI010 引脚输入作为捕获触发。当检测到捕获触发的有效边沿时，将 TM00 的计数值存入 CR000。

（**g**） **16** 位捕获**/**比较寄存器 **010**（**CR010**）

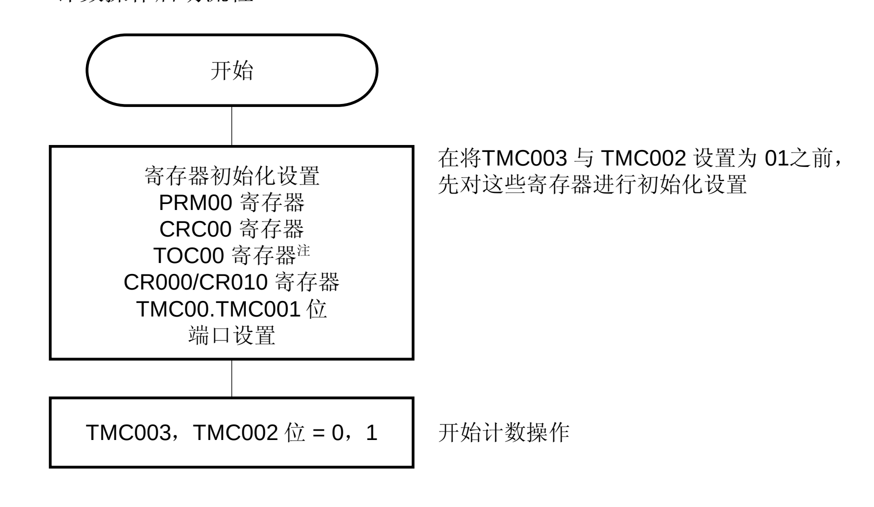
当该寄存器用作比较寄存器并且其值与 TM00 的计数值匹配时，产生中断信号（INTTM010）。但 TM00 的计数值不清除。

当该寄存器用作捕获寄存器，则 TI000 引脚输入作为捕获触发。当检测到捕获触发的有效边沿时，将 TM00 的计数值存入 CR010。

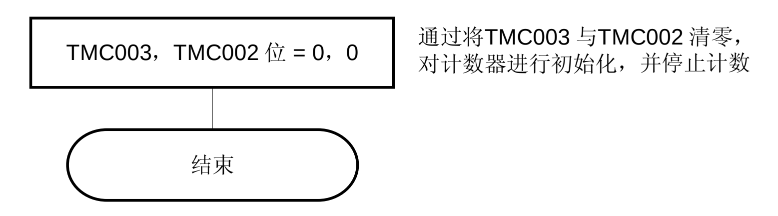
图 6-40. 自由运行定时器模式下软件处理示例



<1> 计数操作启动流程



<2> 计数操作停止流程



注 设置 TOC00 时必须特别注意。详情参见 **6.3（3）16 位定时器输出控制寄存器 00（TOC00）**。

6.4.6 PPG输出操作

当 16 位定时器模式控制寄存器 00（TMC00）的第 3 位和第 2 位（TMC003 与 TMC002）=11 时（TM00 与 CR000 匹配时清零&启动），在由 CR000 设置的周期内从 TO00 引脚输出方波作为 PPG（可编程脉冲发生器）信号。其中脉冲宽度由 CR010 预先设置。

以下是作为 PPG 输出脉冲的周期与占空比。

- 脉冲周期 =（CR000 的设置值 + 1）× 计数时钟周期
- 占空比 =（CR010 的设置值 + 1）/（CR000 的设置值 + 1）

注意事项 若要在操作期间改变占空比（CR010 寄存器的值），参见 6.5.1 TM00 操作期间重写 CR010。

备注 **1.** 如需了解 I/O 引脚的设置，参见 **6.3（6） 端口模式寄存器 3（PM3）**。

2. 要了解如何使能 INTTM000 信号中断，参见**第二十章 中断功能**。

图 **6-41. PPG** 输出操作的框图

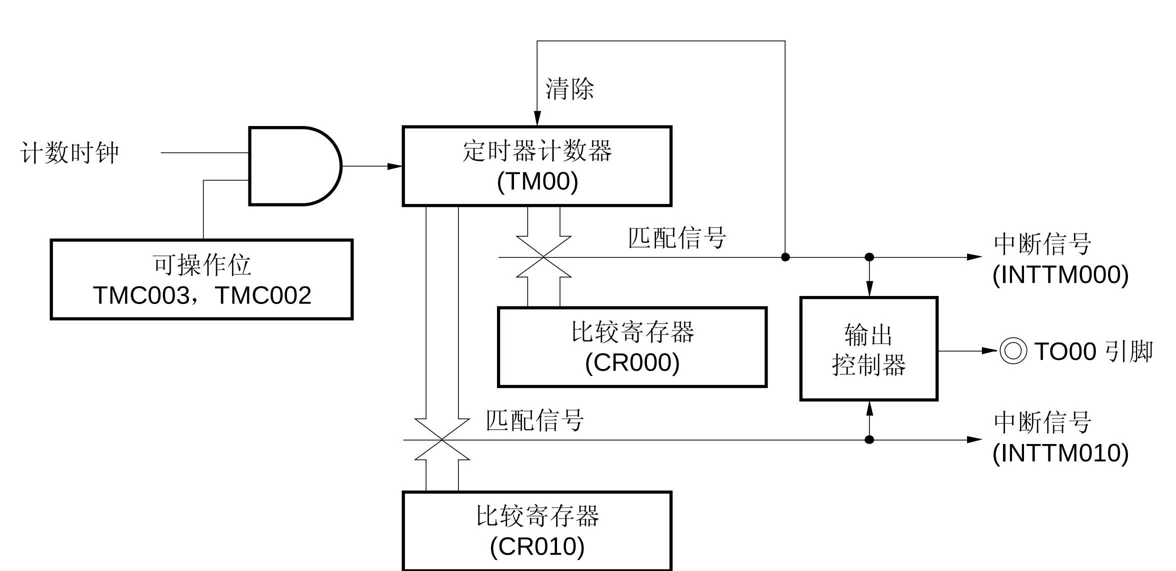
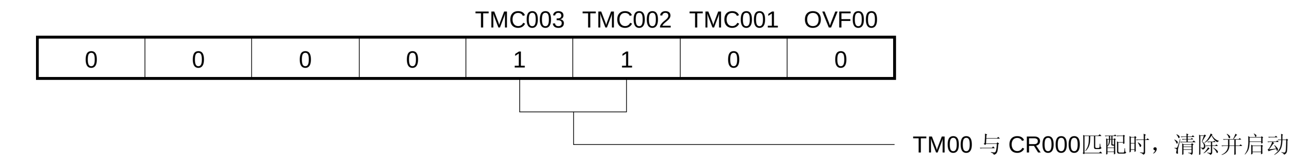
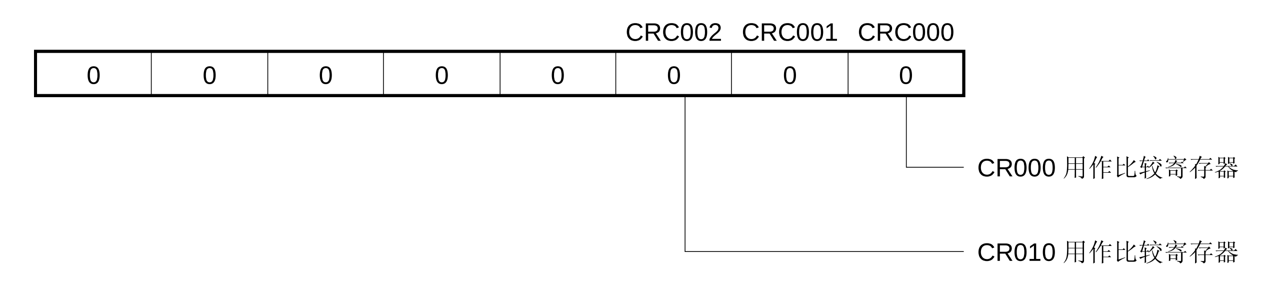


图 **6-42. PPG** 输出操作的寄存器设置示例

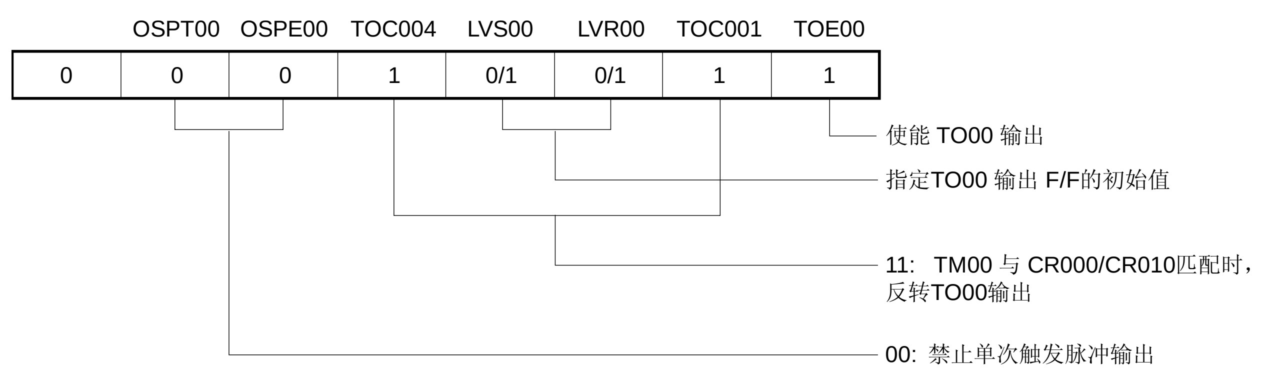
（**a**） **16** 位定时器模式控制寄存器 **00**（**TMC00**）



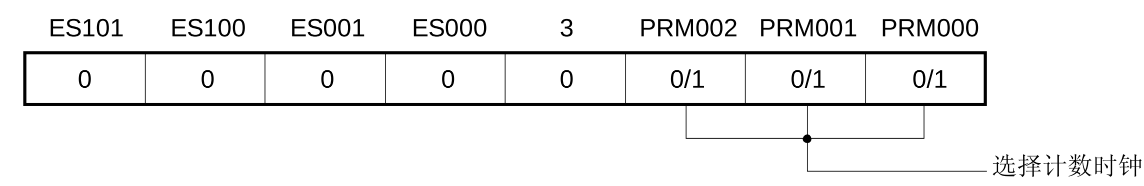
（**b**） 捕获**/**比较控制寄存器 **00**（**CRC00**）



（**c**） **16** 位定时器输出控制寄存器 **00**（**TOC00**）



（**d**） 预分频模式寄存器 **00**（**PRM00**）



（**e**） **16** 位定时器计数器 **00**（**TM00**）

通过读取 TM00，获得计数值。

<R>

（**f**） **16** 位 捕获**/**比较寄存器 **000**（**CR000**）

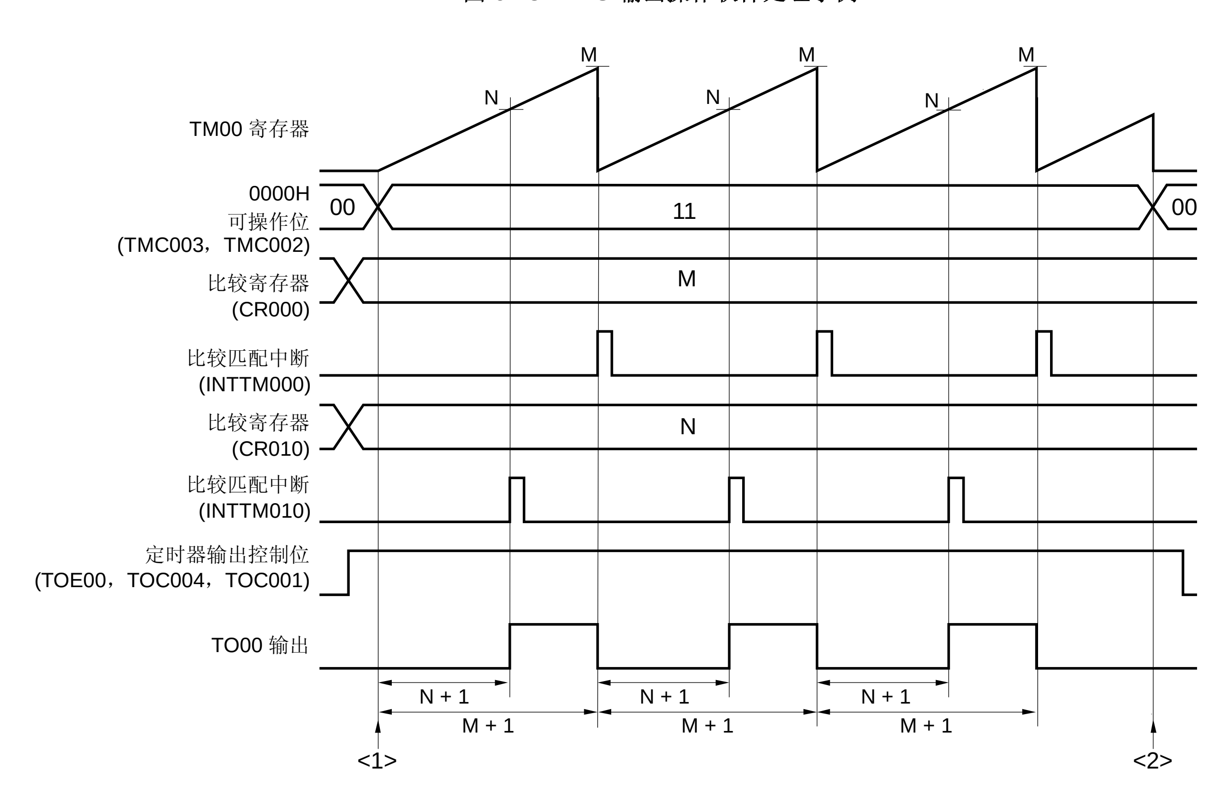
当该寄存器的值与 TM00 的计数值匹配时，产生中断信号（INTTM000）。TM00 的计数值被清除。

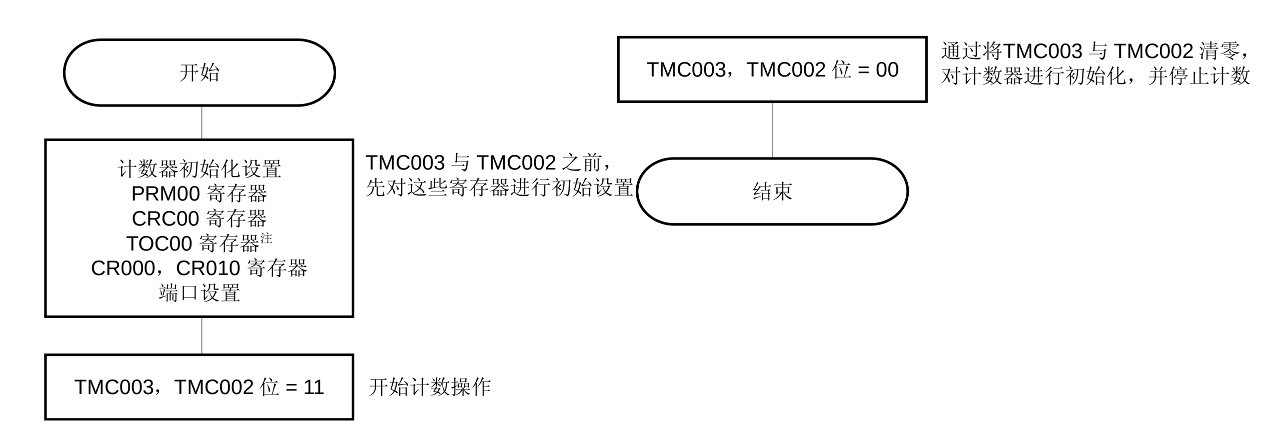
（**g**） **16** 位 捕获**/**比较寄存器 **010**（**CR010**）

当该寄存器的值与 TM00 的计数值匹配时，产生中断信号（INTTM010）。TM00 的计数值不清除。

注意事项 **CR000** 与 **CR010** 的设置值应该满足条件 **0000H ≤ CR010 < CR000 ≤ FFFFH** 。

图 **6-43. PPG** 输出操作软件处理示例





注 设置 TOC00 时必须特别注意。详情参见 **6.3（3）16 位定时器输出控制寄存器 00（TOC00）**。

备注 PPG 脉冲周期 =（M + 1）× 计数时钟周期

PPG 占空比 =（N + 1）/（M + 1）

6.4.7 单次触发脉冲输出操作

将 16 位定时器模式控制寄存器 00（TMC00）的第 3 位和第 2 位（TMC003 与 TMC002）设置为 01（自由运行定时器模式）或 10（通过 TI000 引脚有效边沿进入清零&启动模式），并将 16 位定时器输出控制寄存器 00（TOC00）的第 5 位（OSPE00）设置为 1，可以输出一个单次触发脉冲。

当 TOC00 的第 6 位（OSPT00）被设置为 1 时，或者在定时器操作期间有效边沿输入到 TI000 引脚时，触发 TM00 的清零&启动，并且从 TO00 引脚只输出一次脉冲，该脉冲反映 CR000 与 CR010 之间的差距。

注意事项 1. 当单次触发脉冲输出时，不要再次输入触发（设置 OSPT00=1 或者检测到 TI000 引脚的有效边沿）。若要再次输出单次触发脉冲，应该在当前单次触发脉冲输出完成后再产生触发。

2. 如果仅将 OSPT00 设置为 1 作为单次触发脉冲输出的触发，则不要改变 TI000 引脚的电平或其复用功能端口引脚的电平。否则可能会有脉冲的异常输出。

备注 **1.** 如需了解 I/O 引脚的设置，参见 **6.3（6） 端口模式寄存器 3（PM3）**。

2. 要了解如何使能 INTTM000 信号中断，参见**第二十章 中断功能**。

图 **6-44.** 单次触发脉冲输出操作的框图

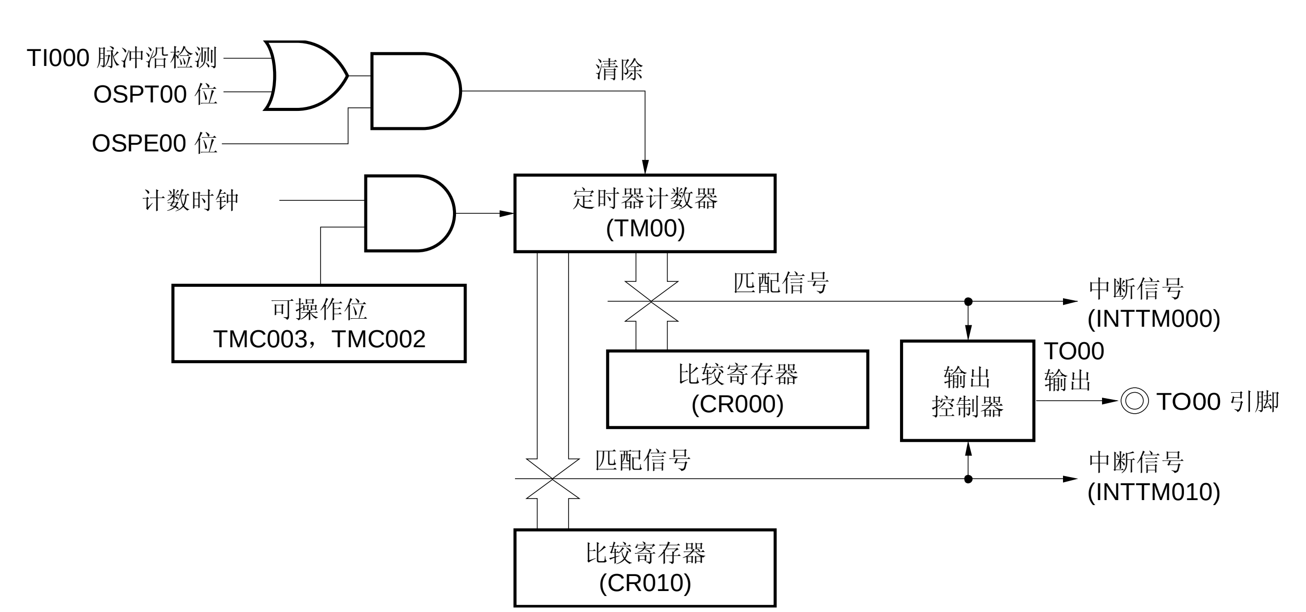
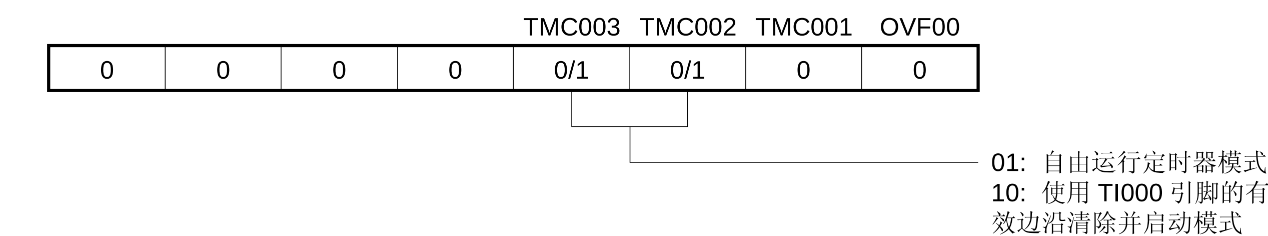
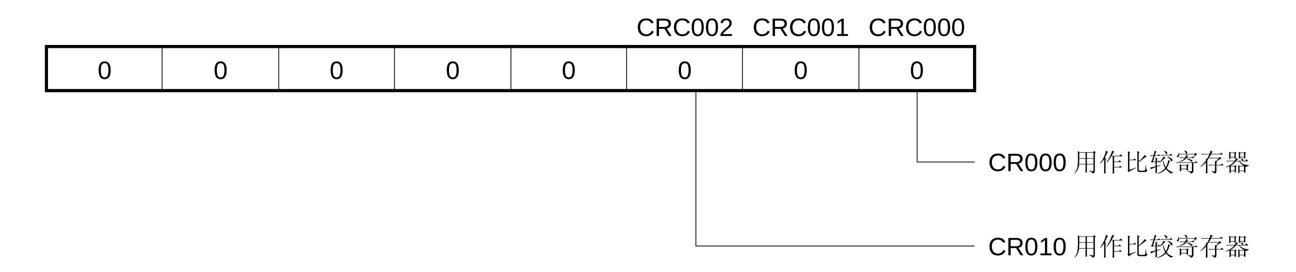


图 **6-45.** 单次触发脉冲输出操作的寄存器设置示例（**1/2**）

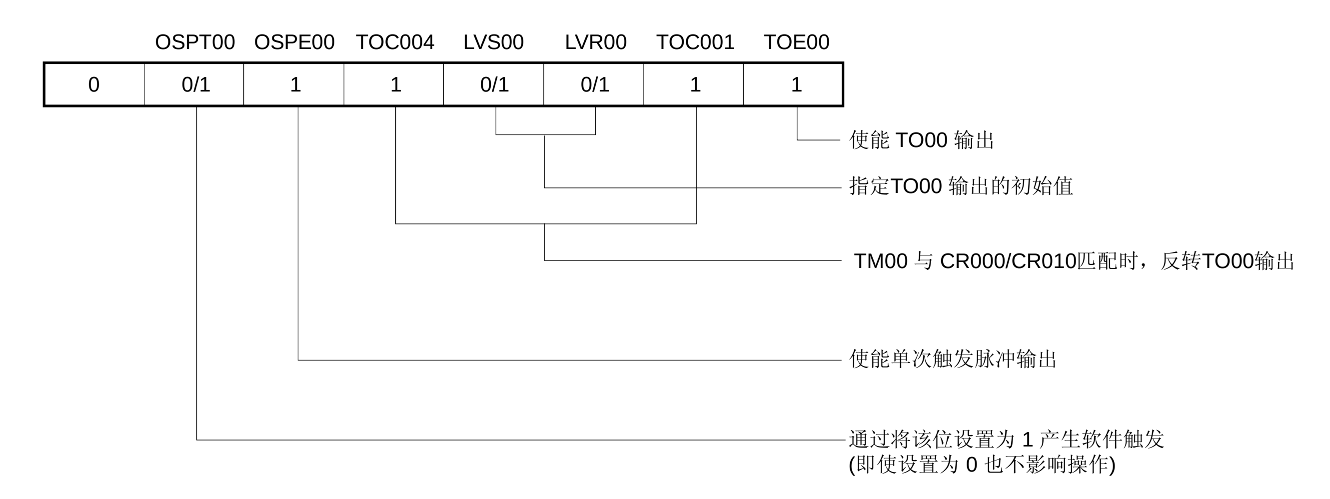
（**a**） **16** 位定时器模式控制寄存器 **00**（**TMC00**）



（**b**） 捕获**/**比较控制寄存器 **00**（**CRC00**）



（**c**） **16** 位定时器输出控制寄存器 **00**（**TOC00**）



（**d**） 预分频模式寄存器 **00**（**PRM00**）

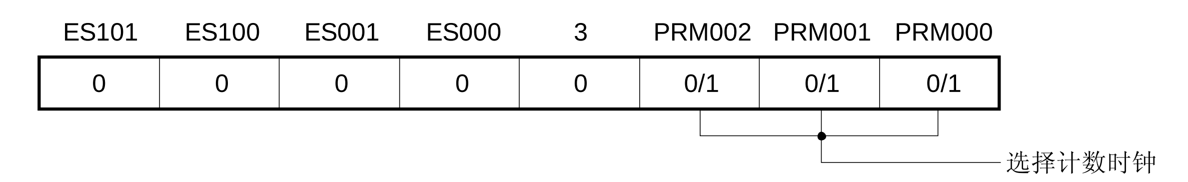


图 6-45. 单次触发脉冲输出操作的寄存器设置示例（2/2）

（e） 16 位定时器计数器 00（TM00）

通过读取 TM00，可获得计数值。

（f） 16 位捕获/比较寄存器 000（CR000）

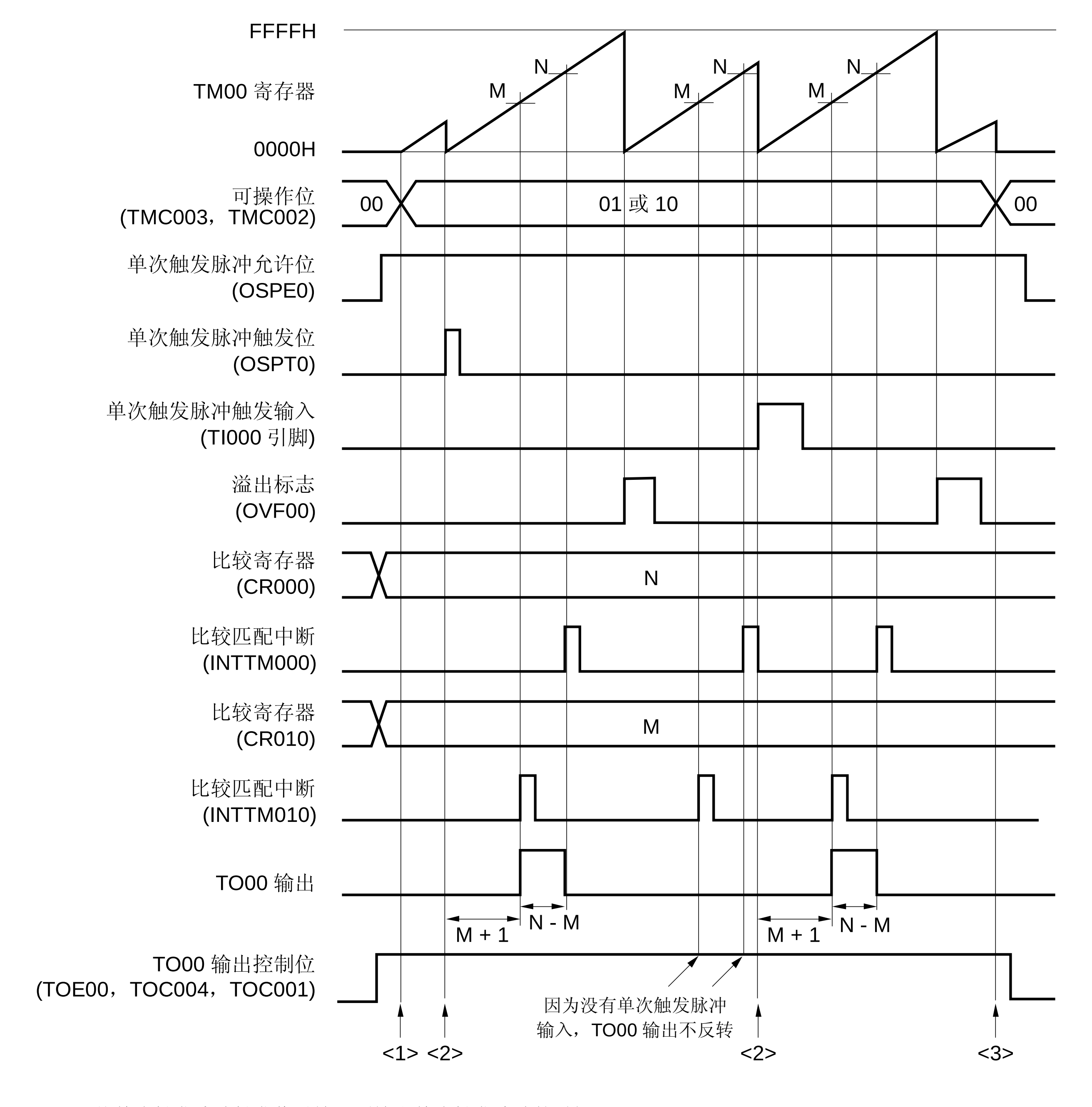
当输出单次触发脉冲时，该寄存器用作比较寄存器。当 TM00 的值与 CR000 的值匹配时，产生中断信号（INTTM000）并反转 TO00 的输出电平。

（g） 16 位捕获/比较寄存器 010（CR010）

当输出单次触发脉冲时，该寄存器用作比较寄存器。当 TM00 的值与 CR010 的值匹配时，产生中断信号（INTTM010）并反转 TO00 的输出电平。

注意事项 不要将 **CR000** 与 **CR010** 的值设置为相同的值。

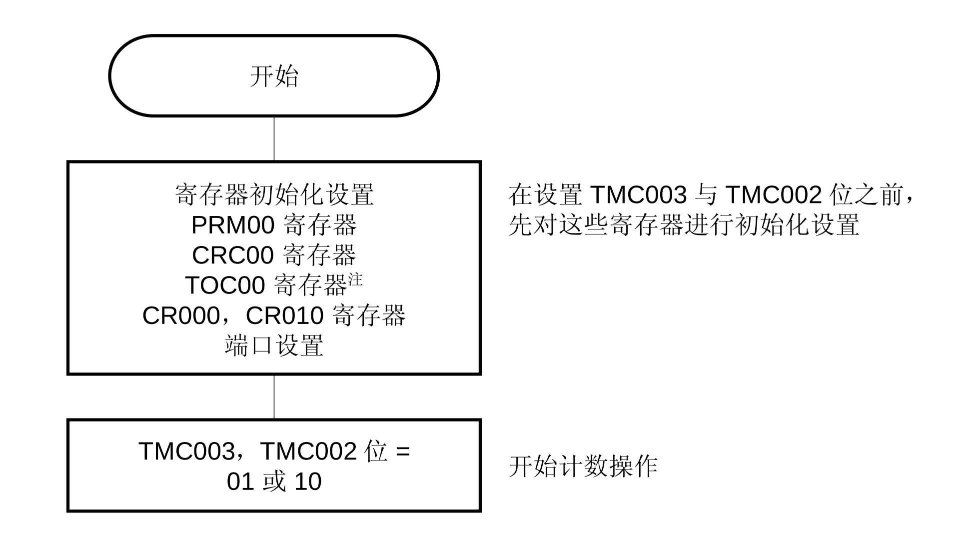
图 6-46. 单次触发脉冲输出操作软件处理示例（1/2）



- 从单次触发脉冲触发信号输入到输出单次触发脉冲的时间
  =（M + 1）× 计数时钟周期
- 单次触发脉冲输出活动电平宽度
  =（N – M）×计数时钟周期

图 6-46. 单次触发脉冲输出操作软件处理示例（2/2）

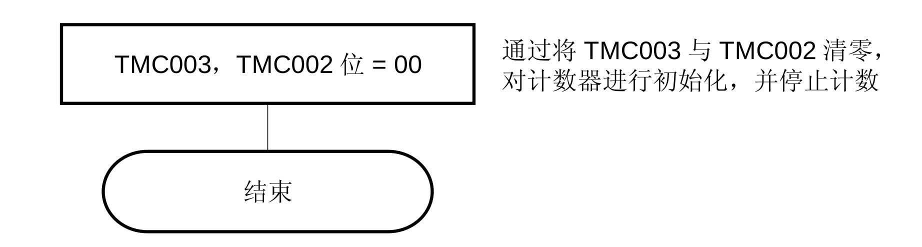
<1> 计数操作启动流程



<2> 单次触发脉冲触发输入流程



<3> 计数操作停止流程



注　设置 TOC00 时必须特别注意。详情参见 **6.3（3）16 位定时器输出控制寄存器 00（TOC00）**。

6.4.8 脉冲宽度测量操作

TM00 可用来测量输入到 TI000 与 TI010 引脚的信号脉冲宽度。

在自由运行定时器模式下操作 16 位定时器/事件计数器 00 或随着输入 TI000 引脚的信号同步重启定时器，可以完成测量。

当产生中断时，读取有效捕获寄存器的值，并测量脉冲宽度。检查 16 位定时器模式控制寄存器 00（TMC00）的第 0 位（OVF00）。如果该位被置位（1），则由软件将其清除为 0。

图 6-47. 脉冲宽度测量（自由运行定时器模式）的框图

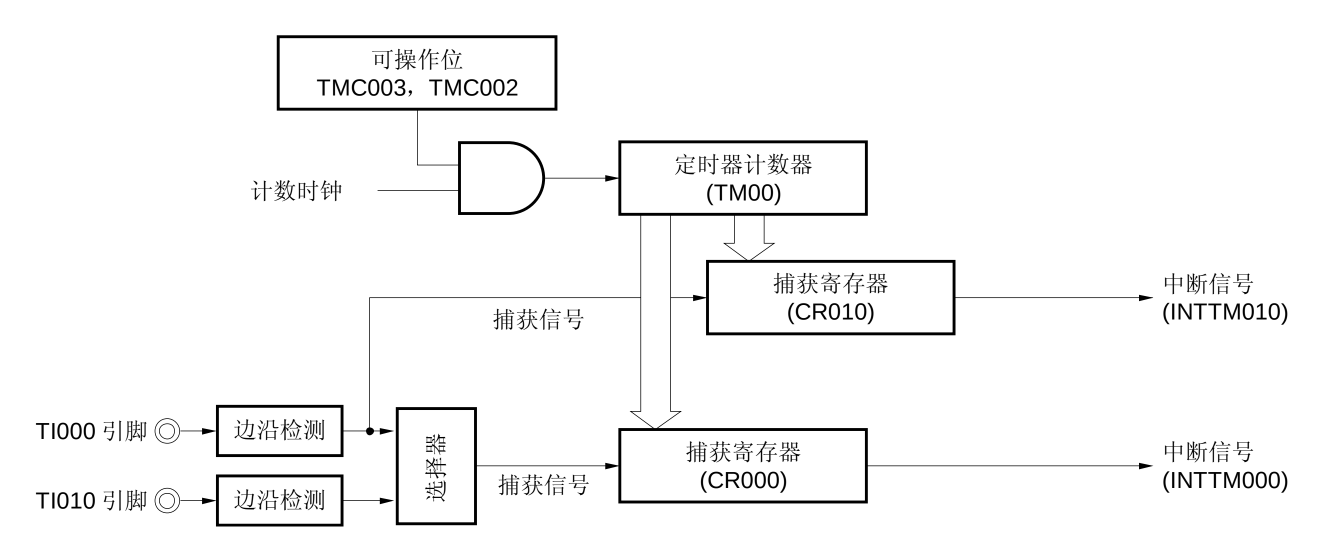
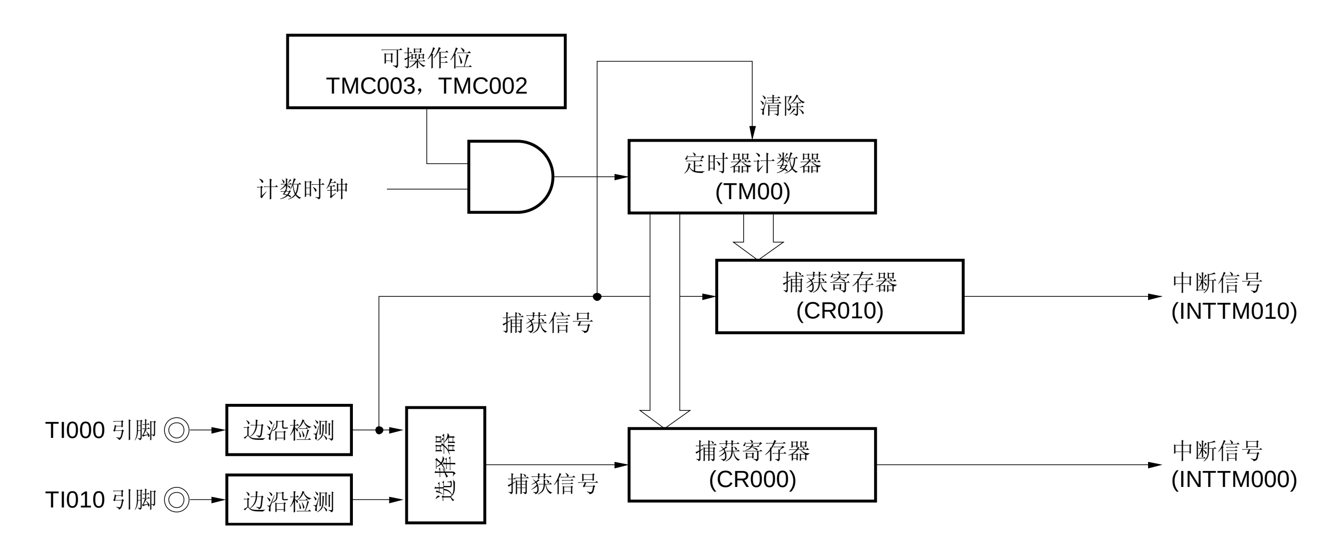


图 6-48. 脉冲宽度测量的框图
（通过输入 TI000 引脚的有效边沿进入清零&启动模式）



可以使用以下三种方法测量脉冲宽度。

- 通过使用 TI000 与 TI010 引脚的两个输入信号测量脉冲宽度（自由运行定时器模式）
- 通过使用 TI000 引脚的一个输入信号测量脉冲宽度（自由运行定时器模式）
- 通过使用 TI000 引脚的一个输入信号测量脉冲宽度（通过输入 TI000 引脚的有效边沿进入清零&启动模式）

注意事项 当测量脉冲宽度时，不要选择 TI000 有效边沿作为计数时钟。

备注 **1.** 如需了解 I/O 引脚的设置，参见 **6.3（6） 端口模式寄存器 3（PM3）**。
2. 要了解如何使能 INTTM000 信号中断，参见**第二十章 中断功能**。

（1） 使用 TI000 与 TI010 引脚的两个输入信号来测量脉冲宽度（自由运行定时器模式）

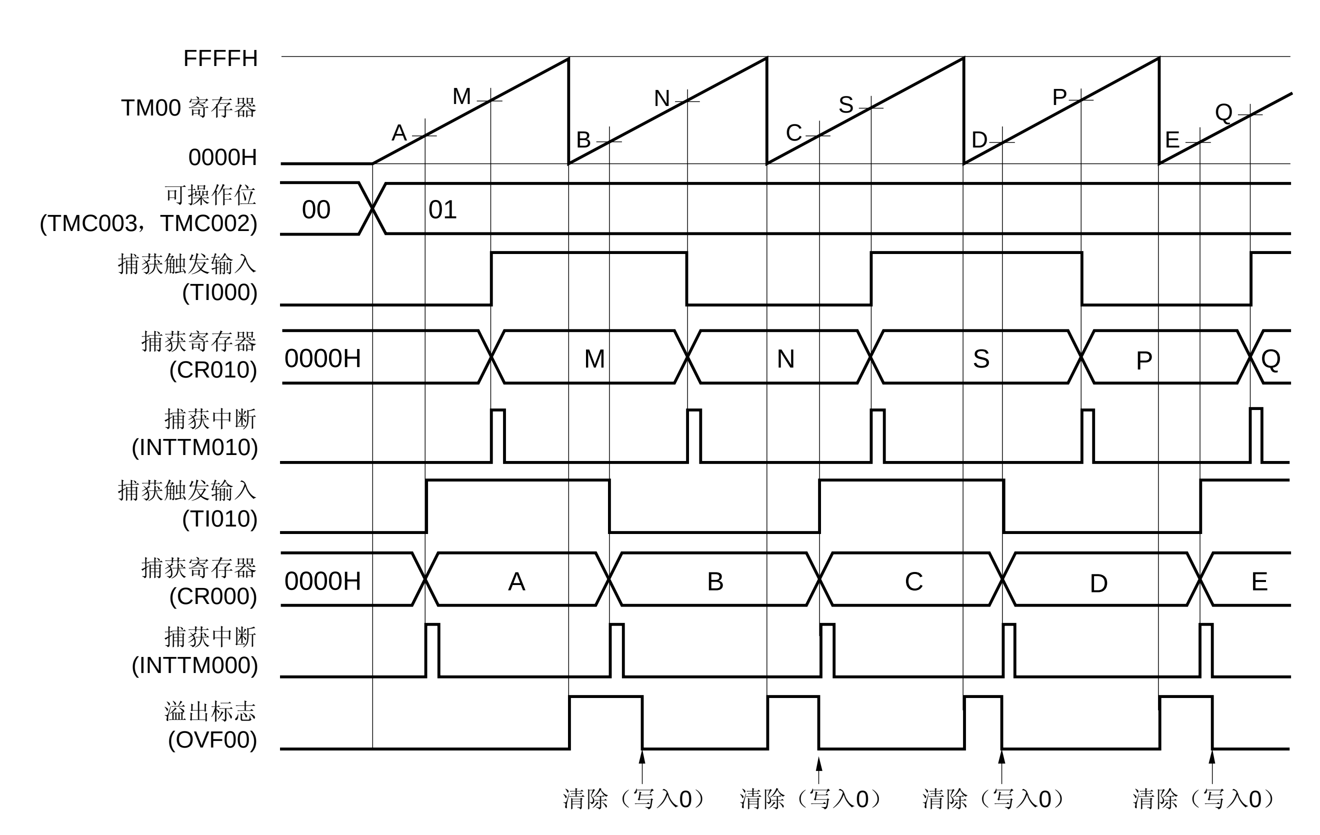
设置自由运行定时器模式（TMC003 与 TMC002 = 01）。当检测到 TI000 引脚的有效边沿时，TM00 的计数值被捕获到 CR010。当检测到 TI010 引脚的有效边沿时，TM00 的计数值被捕获到 CR000。指定检测 TI000 与 TI010 引脚的双沿。

使用这种测量方法，要从每个输入信号的脉冲沿时捕获到的计数值中减去上次的计数值。因此要将上次的捕获值提前存入一个单独的寄存器中。

如果产生溢出，则直接从当前捕获值减去上次捕获值得到的值为负数，并因此产生借位（程序状态字（PSW）的第 0 位（CY）被置 1）。如果发生这种情况，可以忽略 CY 并将计算结果作为脉冲宽度。此外，将 16 位定时器模式控制寄存器 00（TMC00）的第 0 位（OVF00）清除为 0。

图 6-49. 脉冲宽度测量时序示例 （1）

• TMC00 = 04H，PRM00 = F0H，CRC00 = 05H



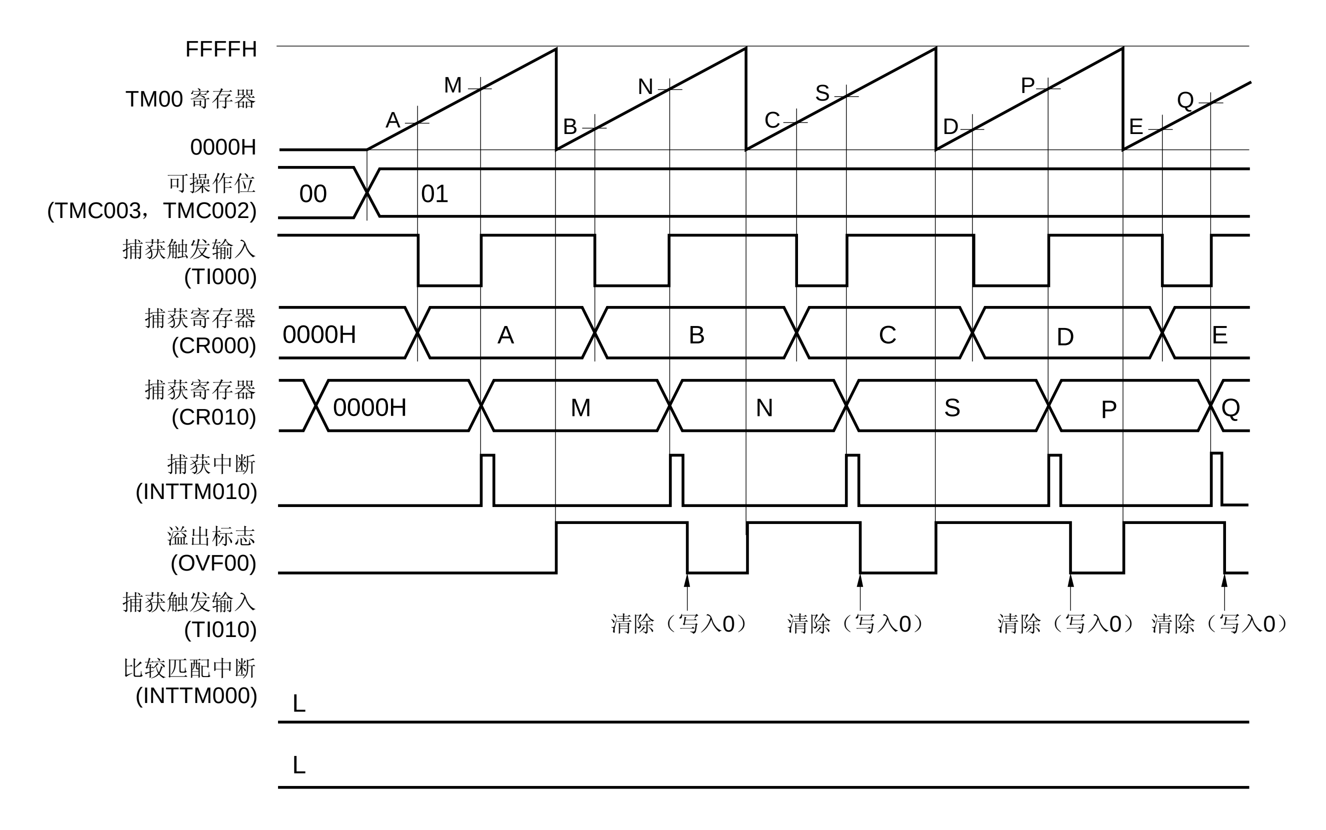
（2） 使用 TI000 引脚的一个输入信号来测量脉冲宽度（自由运行模式）

设置自由运行定时器模式（TMC003 与 TMC002 = 01）。在 TI000 引脚检测到有效边沿的反向沿时，将 TM00 的计数值捕获到 CR000。当检测到 TI000 引脚的有效边沿时，将 TM00 的计数值捕获到 CR010。

使用这种测量方法，当测量从一个脉冲沿到另一个脉冲沿的宽度时，将数值存入单独的两个捕获寄存器中。因此，不必保存捕获值。将两个捕获寄存器的值相减，可以计算得到高电平宽度、低电平宽度和周期。

如果产生溢出，则直接从一个捕获值减去另一个捕获值得到的值为负数，并因此产生借位（程序状态字（PSW）的第 0 位（CY）被置 1）。如果发生这种情况，可以忽略 CY 并将计算结果作为脉冲宽度。此外，将 16 位定时器模式控制寄存器 00（TMC00）的第 0 位（OVF00）清除为 0。

图 6-50. 脉冲宽度测量时序示例（2）

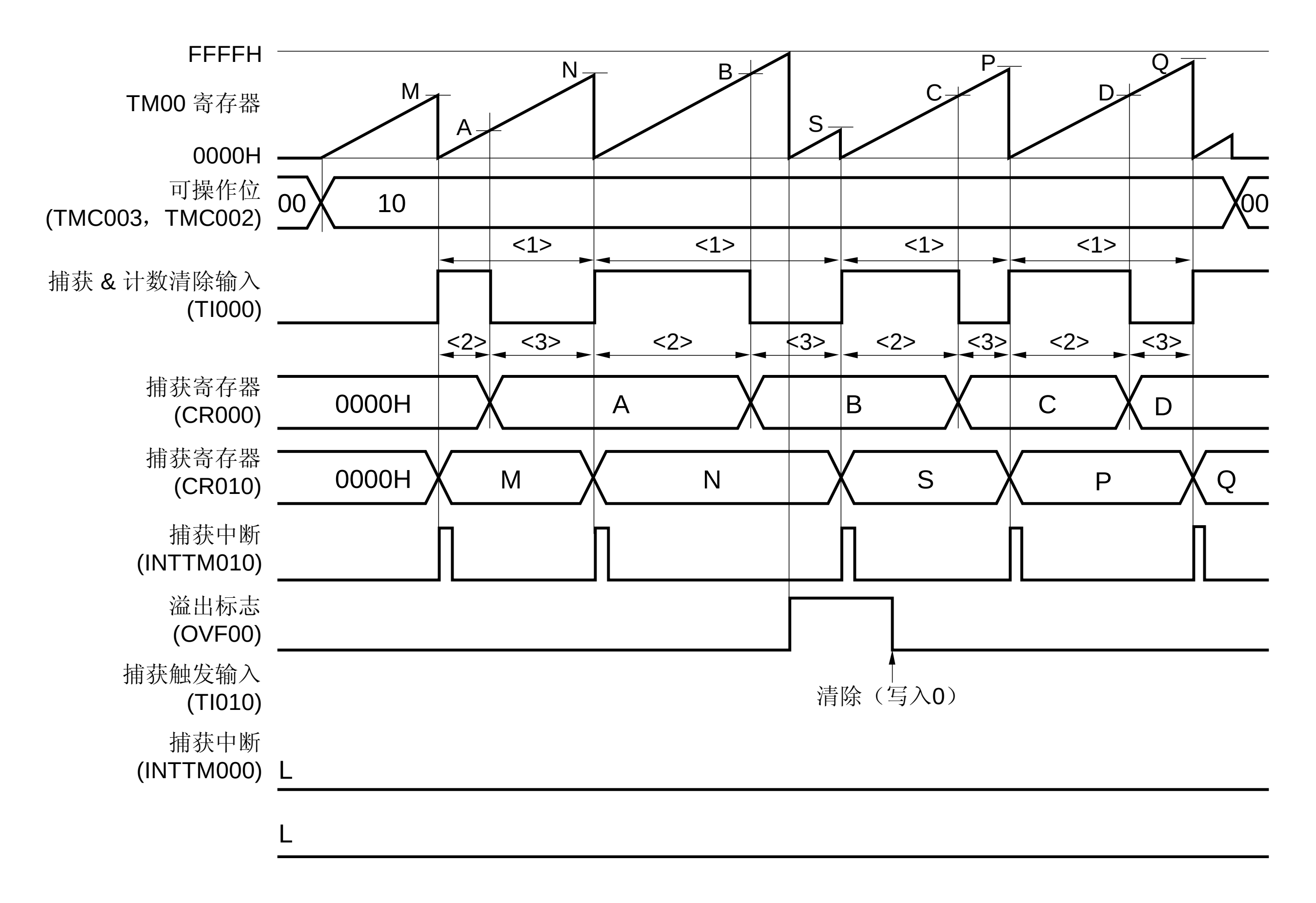


（3） **使用 TI000 引脚的一个输入信号来测量脉冲宽度（通过输入 TI000 引脚的有效边沿进入清零&启动模式）**

设置通过输入 TI000 引脚的有效边沿进入清零&启动模式（TMC003 与 TMC002 = 10）。在 TI000 引脚检测到有效边沿的反向沿时，将 TM00 的计数值捕获到 CR000。当检测到 TI000 引脚的有效边沿时，将 TM00 的计数值捕获到 CR010，并将 TM00 清除（0000H）。因此，如果 TM00 没有溢出，则在 CR010 中存入一个周期值。

如果产生溢出，则将 CR010 的值加上 10000H 作为一个周期。此外，将 16 位定时器模式控制寄存器 00（TMC00）的第 0 位（OVF00）清除为 0。

图 6-51. 脉冲宽度测量时序（3）



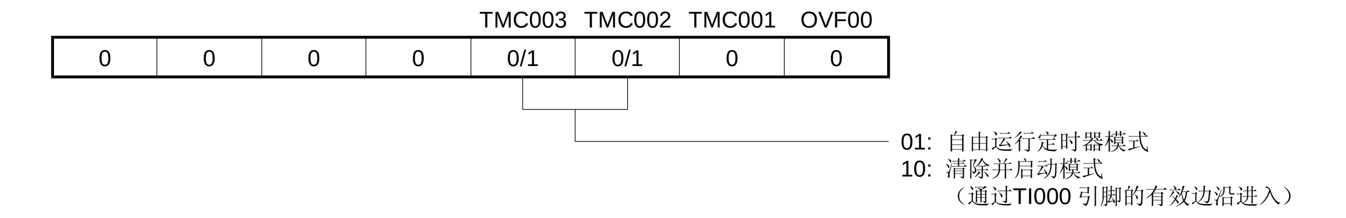
<1> 脉冲周期 = （10000H × OVF00 位被设置为 1 的次数 +CR010 的捕获值）× 计数时钟周期

<2> 高电平脉冲宽度 = （10000H × OVF00 位被设置为 1 的次数+ CR000 的捕获值）×计数时钟周期

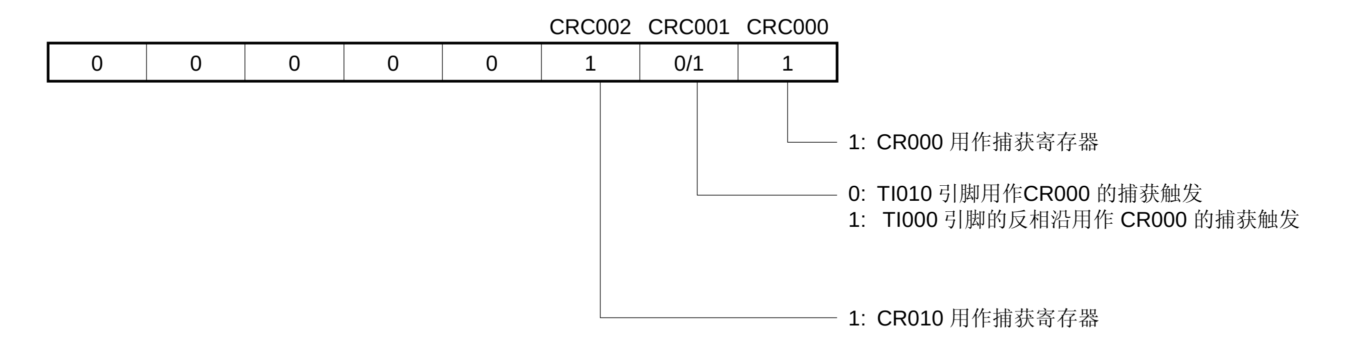
<3> 低电平脉冲宽度 = （脉冲周期 – 高电平脉冲宽度）

图 6-52. 脉冲宽度测量的寄存器设置示例（1/2）

（a） 16 位定时器模式控制寄存器 00（TMC00）



（b） 捕获/比较控制寄存器 00（CRC00）



（c） 16 位定时器输出控制寄存器 00（TOC00）

	OSPT00	OSPE00	TOC004	LVS00	LVR00	TOC001	TOE00
0	0	0	0	0	0	0	0

（d） 预分频模式寄存器 00（PRM00）

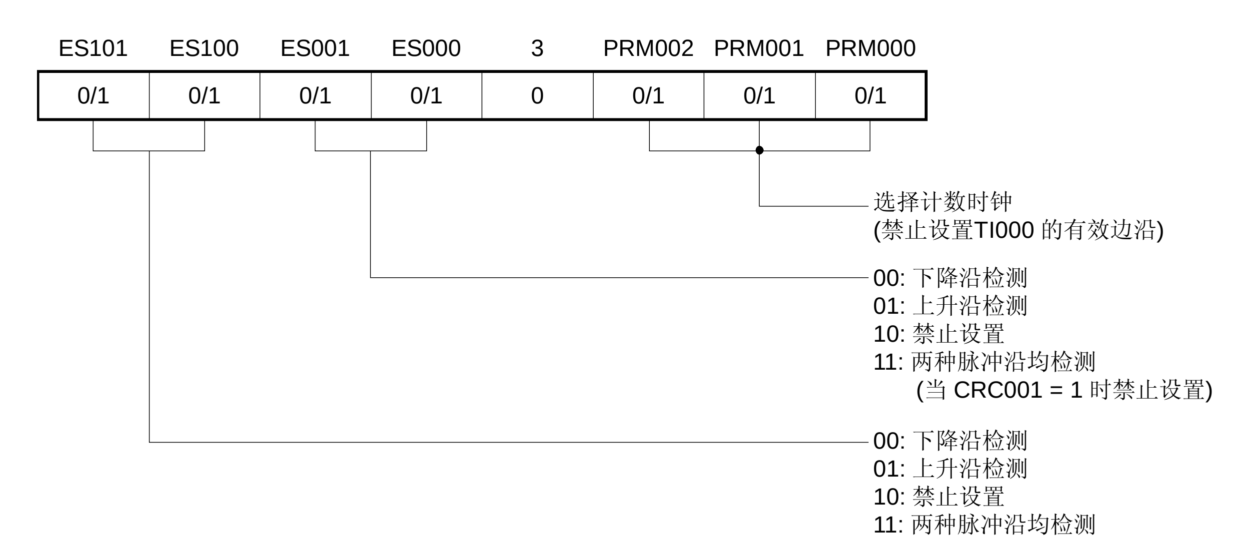


图 6-52. 脉冲宽度测量的寄存器设置示例（2/2）

（e） 16 位定时器计数器 00（TM00）

通过读取 TM00，可获得计数值。

（f） 16 位 捕获/比较寄存器 000（CR000）

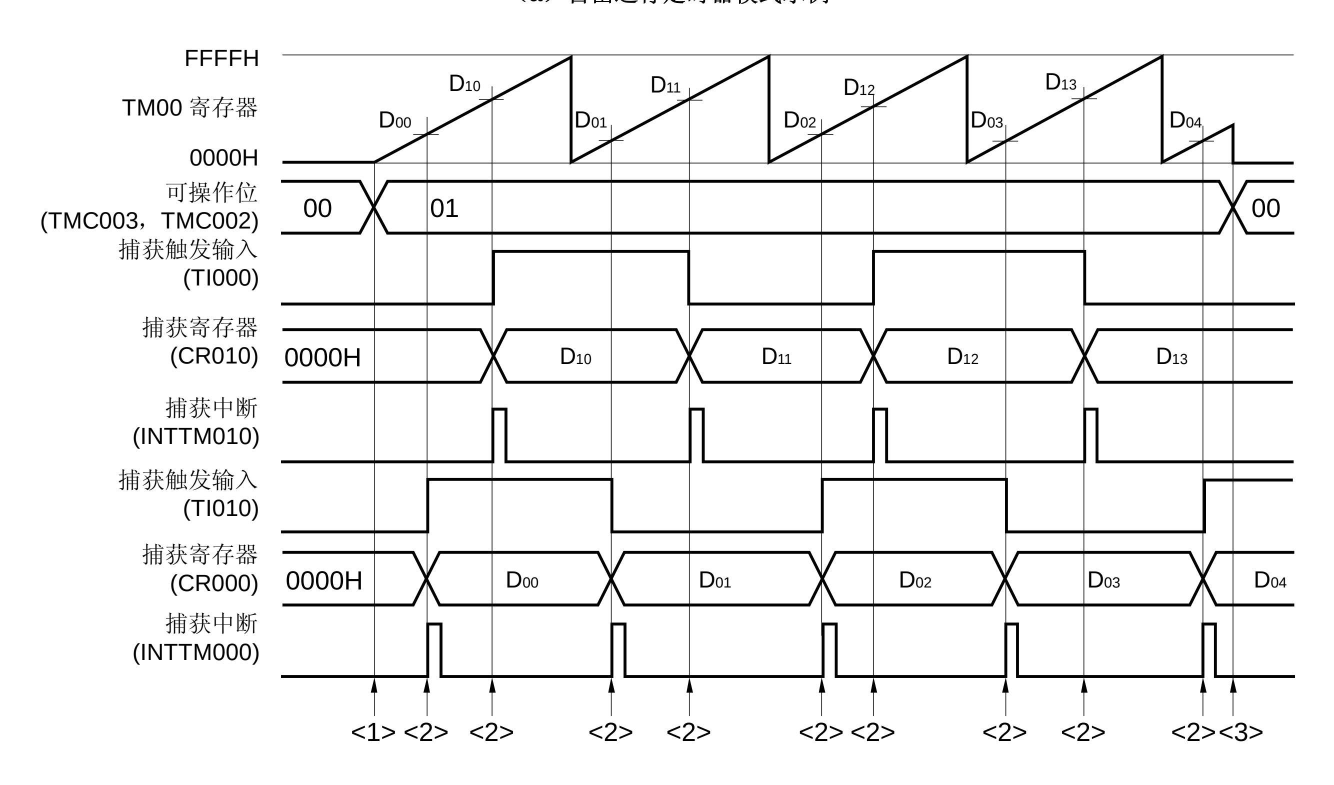
该寄存器用作捕获寄存器。选择 TI000 或 TI010 引脚作为捕获触发。当检测到捕获触发的指定边沿时，将 TM00 的计数值存入 CR000。

（g） 16 位 捕获/比较寄存器 010（CR010）

该寄存器用作捕获寄存器。TI000 引脚的输入信号被用作捕获触发。当检测到捕获触发时，将 TM00 的计数值存入 CR010。

图 6-53. 脉冲宽度测量的软件处理示例（1/2）

（**a**）自由运行定时器模式示例



（**b**）通过输入 **TI000** 引脚的有效边沿进入清零**&**启动模式示例

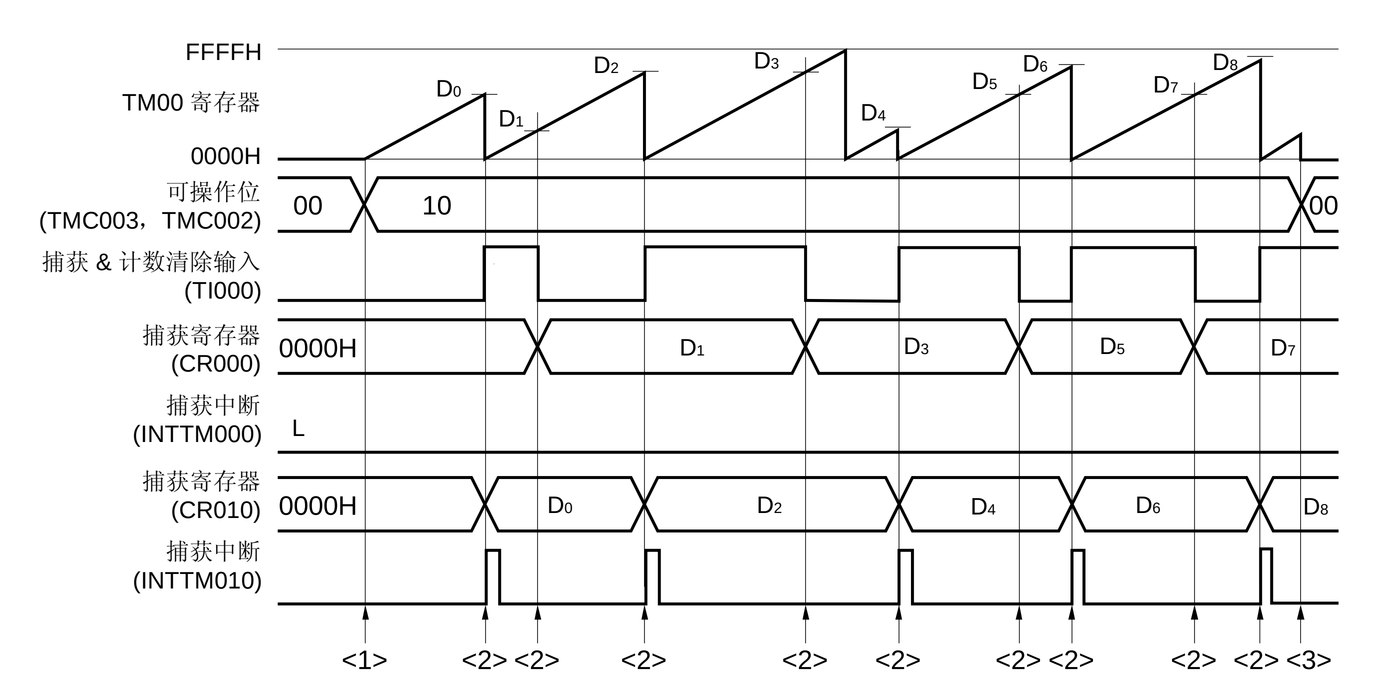
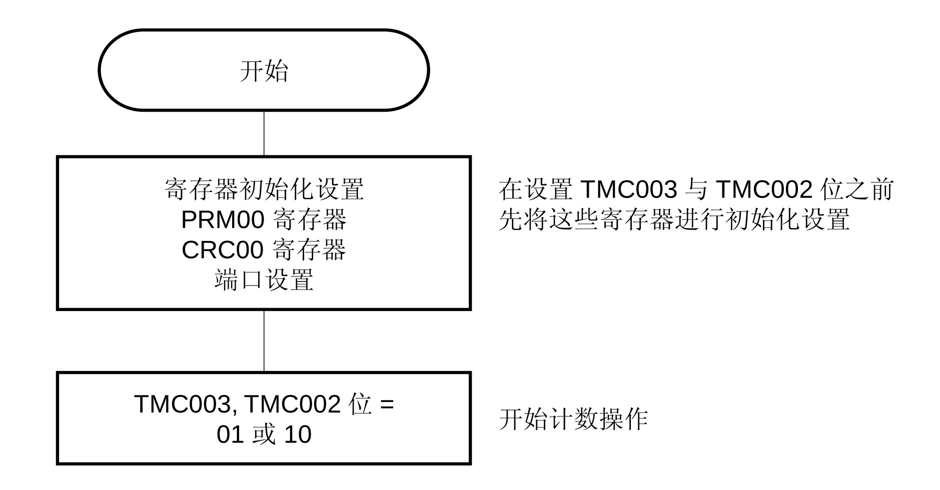
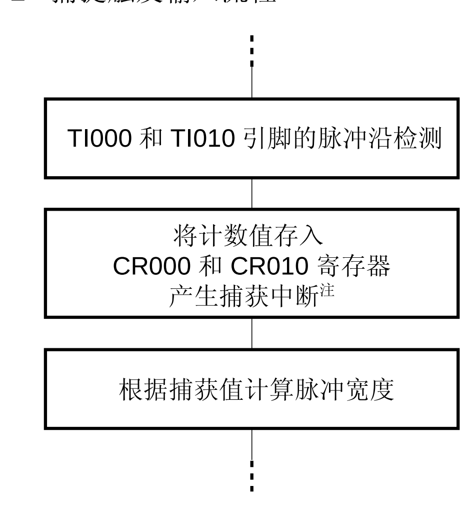


图 6-53. 脉冲宽度测量的软件处理示例（2/2）

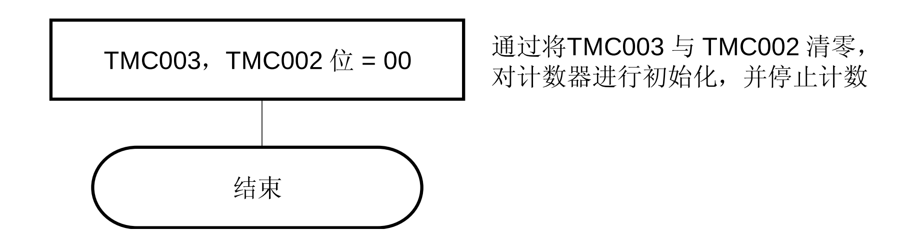
<1> 计数操作启动流程



<2> 捕捉触发输入流程



<3> 计数操作停止流程



注 当选择 TI000 引脚输入的反向边沿作为 CR000 的有效边沿时，不产生捕获中断信号（INTTM000）。

6.4.9 外部 24 位事件计数器操作

通过级联 16 位定时器/事件计数器 00 和 8 位定时器/时间计数器 52，16 位定时器/事件计数器 00 可用于外部 24 位事件计数器功能，使用 8 位定时器/事件计数器 52 的外部事件计数器功能。

通过 8 位定时器 52（TM52）对输入到 TI52 引脚的外部时钟脉冲数量进行计数，并通过 16 位定时器计数器 00（TM00）对 TM52 计数值与 8 位定时器比较寄存器 52（CR52=FFH [注]）匹配时输出的信号进行计数，可以作为外部 24 位事件计数器操作

当 16 位定时器/事件计数器 00 用于外部 24 位事件计数器时，可通过 8 位定时器计数器 H2 输出来控制外部事件输入的使能。

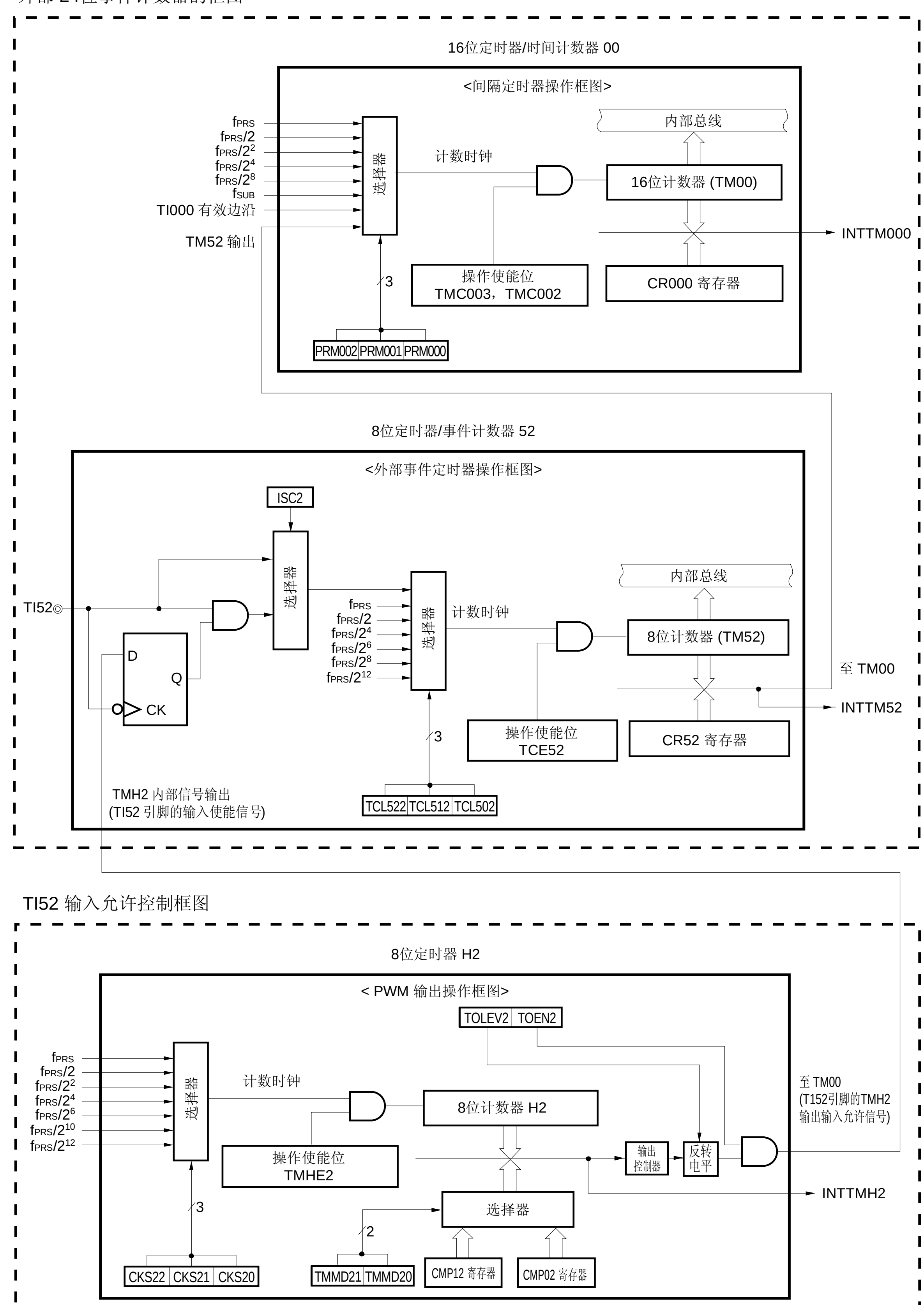
输入到 TI52 引脚的有效边沿可由 8 位定时器计数器 52（TM52）的定时器时钟选择寄存器 52（TCL52）指定。将输入切换控制寄存器（ISC）的第 2 位（ISC2）设置为“1”，则可以通过 8 位定时器计数器 H2 输出来控制 TM52 外部事件输入的输入使能。

设置 16 位定时器/事件计数器 00 的预分频模式寄存器 00（PRM00）的第 2，1 和 0 位（PRM002，PRM001 和 PRM000）为“1”，“1”和“1”，并设置 16 位定时器模式控制寄存器 00（TMC00）的第 3 和 2 位（TMC003 和 TMC002）为“1”和“1”（TM00 和 CR000 匹配时进入计数清零&启动模式），计数操作开始，8 位定时器 52 的输出用作计数时钟。TM00 计数值与 16 位定时器比较寄存器 000（CR000）的值匹配时，TM00 被清除为“0”，并且产生一个中断请求信号（INTTM000）。

随后，在 TM00 和 CR000 的值匹配时产生 INTTM000。

注　当 16 位定时器/事件计数器 00 被用作外部 24 位事件计数器操作时，8 位定时器比较寄存器 52（CR52）的值必须设置为 FFH。而且，TM52 中断请求信号（INTTM52）必须被屏蔽（TMMK52 = 1）。

图 6-54. 外部 24 位事件计数器的框图



设置

<1> TM00 和 TM52 的各种模式的设置。

（a） TM00 设置为间隔定时器。选择 TM52 输出作为计数时钟。

- TMC00： 设置操作禁止。
  （TMC00 = 00000000B）
- CRC00： 设置作为比较寄存器。
  （CRC00 = 000000x0B，x = don't care）
- TOC00： 设置当 CR000 和 TM00 匹配时禁止 TO00 引脚输出。
  （TOC00 = 00000000B）
- PRM00： TM52 输出被选择作为计数时钟。
  （PRM00 = 00000111B）
- CR000： 设置比较值为 FFFFH。
  如果比较值设置为 M，TM00 将向上计数到 M。
- CR010： 通常不使用 CR010，但是，当 CR010 的设置值和 TM00 值匹配时会产生一个比较匹配中断（INTTM010）。因此，通过中断屏蔽标志（TMMK010）屏蔽该中断请求。

（b） 设置 TM52 作为外部事件计数器。

- TCL52： TI52 引脚输入的边沿选择
  TI52 引脚的下降沿 → TCL52 = 00H
  TI52 引脚的上升沿 → TCL52 = 01H
- CR52： 设置比较寄存器的值为 FFH。
- TMC52： 计数操作停止。
  （TMC52 = 00000000B）
- TMIF52： 清除该寄存器。

注意事项 **当 16 位定时器/事件计数器 00 被用作外部 24 位事件计数器操作时，必须屏蔽 INTTM52（TMMK52 = 1）。并且，比较寄存器 52（CR52）的值必须设置为 FFH。**

（c） 为 TI52 引脚将 TMH2 设置为输入允许宽度调整模式（PWM 模式）。[注]

- TMHMD2： 计数操作停止，选择计数时钟，模式设置到输入允许宽度调整模式（PWM 模式），定时器输出电平默认值设置为高电平，定时器输出设置为使能（TMHMD2 = 0xxx1011B，x = 根据使用条件设置）。
- CMP02： 比较值（N），频率设置
- CMP12： 比较值（M），占空比设置
  备注 00H ≤ CMP12（M）< CMP02（N）≤ FFH
- ISC2： 设置 ISC2 = 1（TI52 引脚输入使能控制）

注 如果不控制 TI52 引脚的输入使能，无需该设置。

<2> TM00，TM52 和 TMH2 计数操作开始。定时器操作必须按照下列过程开始。

（a） 设置 TMC003 和 TMC002 位为 1 和 1，启动 TM00 计数器操作。

（b） 设置 TCE52 为 1，启动 TM52 计数器操作。

（c） 设置 TMHE2 为 1，启动 TMH2 计数器操作。[注]

注 如果不控制 TI52 引脚的输入使能，无需该设置。

<3> 当 TM52 和 CR52（= FFH）的值匹配时，TM52 被清除为 00，并且匹配信号使 TM000 开始向善计数。然后，当 TM000 和 CR000 匹配时，TM00 被清为 0000H，并产生一个匹配中断信号（INTTM000）。
如果TI52 引脚的输入使能被控制，在TMH2 中断请求信号（INTTMH2）的中断服务中读取TM52、TM00 计数值和TMIF52，可以测量TI52 引脚在输入使能期间的外部事件计数值。

图 6-55. 外部 24 位事件计数器的操作时序

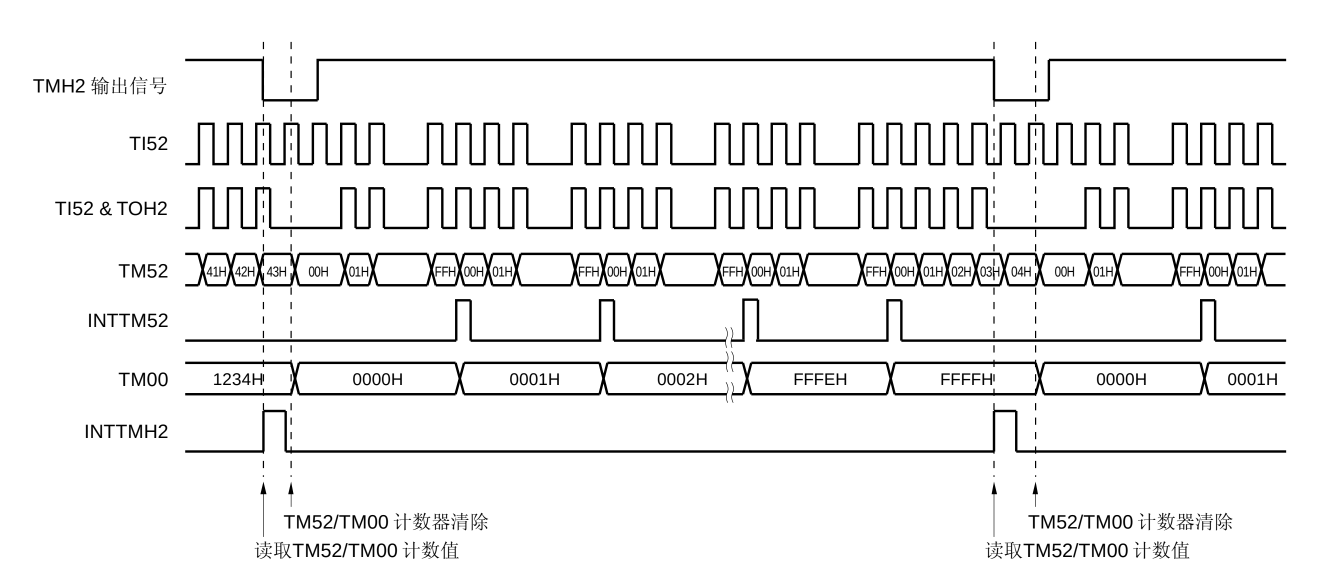
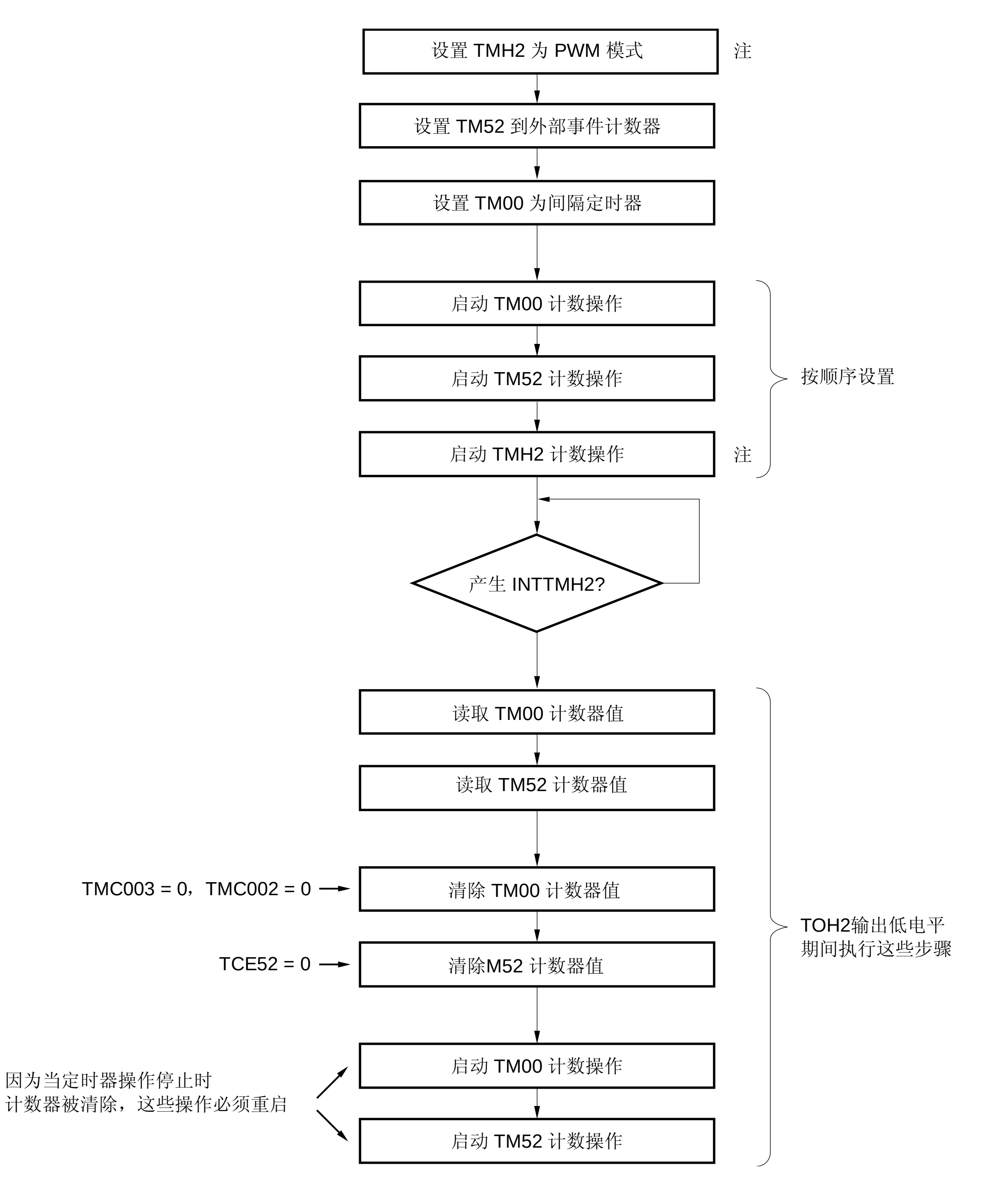


图 6-56. 外部 24 位事件计数器的操作流程图



注 如果不控制 TI52 引脚的输入使能，无需该设置。

6.4.10 外部 24 位事件计数器的注意事项

（1） 8 位定时器计数器 H2 输出信号

8 位定时器 H2 的输出电平控制（默认值），用于控制 TI52 引脚的输入使能，必须设置为高电平（TOLEV2 = 1）。此后，当 TI52 引脚输入使能信号被禁止（TMH2 输出：低电平）时产生一个中断请求信号（INTTMH2），并且可以在该中断的中断服务中读取 TM52 和 TM00 计数值（= 输入使能期间的外部事件计数值）。

请注意，在 8 位定时器 H2 操作已经通过该设置（TOLEV2 = 1）使能（TMHE2 = 1）之后，TI52 引脚的输入使能信号是高电平（使能状态），直到 TMH2 和 CMP02 寄存器的值匹配。

（**2**） **TI52** 引脚输入使能控制的注意事项

TI52 引脚的输入使能控制信号（TMH2 输出信号）与 TI52 引脚输入时钟同步，如图 **6-54 外部 24 位事件计数器的框图**和 **图 6-55 外部 24 位事件计数器的操作时序**中的描述。因此，当计数器作为外部事件计数器操作时，可能会导致一个计数的错误。

（**3**） 外部 **24** 位事件计数器操作期间 **16** 位定时器**/**事件计数器 **00** 向上计数的注意事项

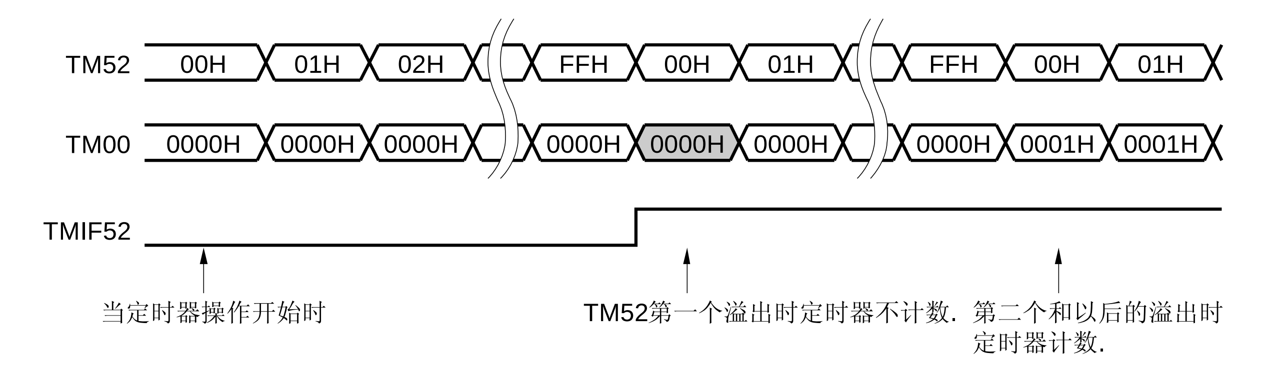
16 位定时器/事件计数器 00 具有内部同步电路，当开始操作时可以消除噪声，操作开始后的并不对第一个时钟立即计数。

通过设置 16 位定时器/事件计数器 00 和 8 位定时器/事件计数器 52 作为高位和低位定时器，并将它们级联，该计数器作为 24 位计数器使用，作为低位定时器的 8 位定时器/事件计数器 52 的中断请求标志必须按照下面的描述进行检查，以便正确读出 24 位计数值。

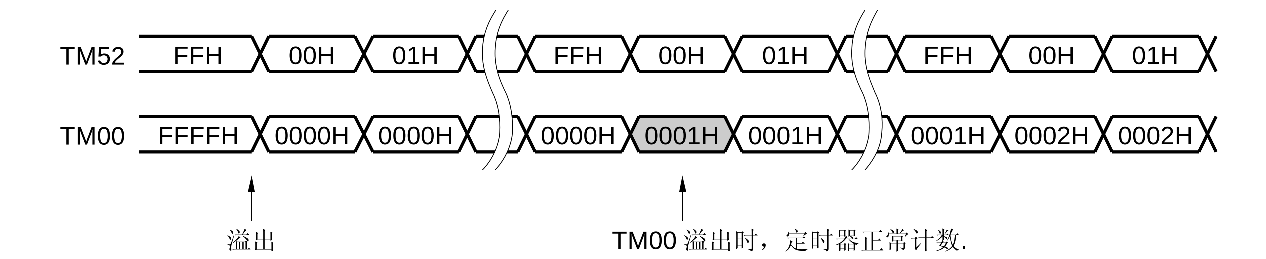
- 如果读取 TM52 和 TM00 时，TMIF52 = 1：
  TM00 实际的计数值是“TM00 的读取值 + 1”。
- 如果读取 TM52 和 TM00 时，TMIF52 = 0：
  读取的值是正确的值。

仅当操作开始后，16 位定时器/事件计数器 00 会出现这种现象。当 16 位定时器/事件计数器 00 溢出且从 0000H 重启计数时不发生计数延迟，因为已经实现同步。

<操作开始时>



<高位定时器的溢出>



6.5 TM00 的特殊用途

6.5.1 TM00 操作期间重写CR010

原则上，在 TM00 操作时（TMC003 与 TMC002 不等于 00），当 CR000 与 CR010 被用作比较寄存器，则禁止在 78K0/LE3 中重写它们的值。

但是，如果 CR010 用于 PPG 输出且占空比已被改变（当 CR010 的值被设置为一个小于或大于当前值的值时，在 CR010 的值与 TM00 的值匹配后或在 CR000 的值与 TM00 的值匹配后，立即重写 CR010 的值。如果在 CR010 的值与 TM00 的值匹配前或在 CR000 的值与 TM00 的值匹配前，立即重写 CR010，则可能会执行异常操作）时，即使 TM00 正在操作，也可以按下列过程改变 CR010。

改变 CR010 值的过程

<1> 禁止中断 INTTM010（TMMK010 = 1）.
<2> 禁止当 TM00 的值与 CR010 的值匹配时反转定时器输出（TOC004 = 0）。
<3> 改变 CR010 的值
<4> 等待 TM00 计数时钟的一个周期
<5> 使能当 TM00 的值和 CR010 的值匹配时反转定时器输出（TOC004 = 1）。
<6> 清除 INTTM010 的中断标志（TMIF010 = 0）。
<7> 使能中断 INTTM010（TMMK010 = 0）.

备注 如需了解 TMIF010 和 TMMK010，参见 **第二十章 中断功能**。

6.5.2 设置LVS00 和LVR00

（1） LVS00 和 LVR00 的使用

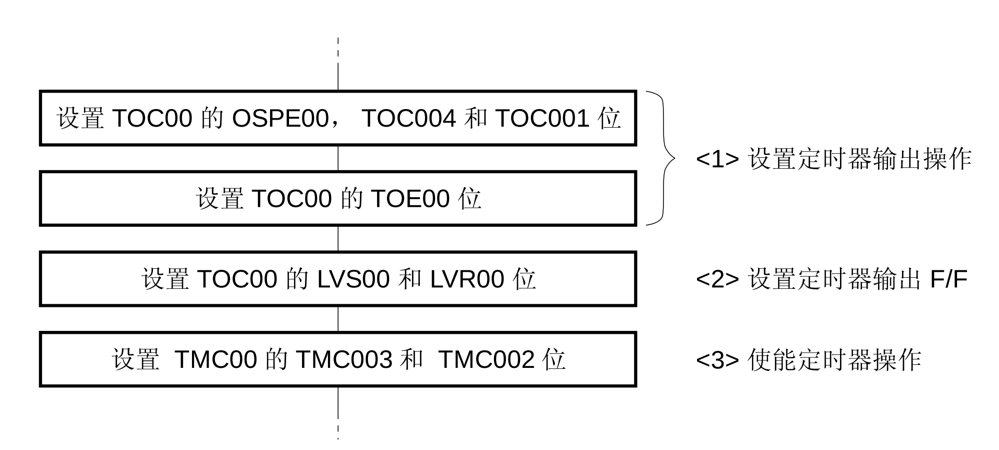
LVS00 与 LVR00 用于设置 TO00 输出的默认值，并且无需使能定时器操作（TMC003 与 TMC002 = 00）就可以反转定时器输出。当不需要软件控制时，将 LVS00 与 LVR00 清除为 00（默认值： 低电平输出）。

LVS00	LVR00	定时器输出状态
0	0	无变化（低电平输出）
0	1	清除（低电平输出）
1	0	置位（高电平输出）
1	1	禁止设置

（2） 设置 LVS00 和 LVR00

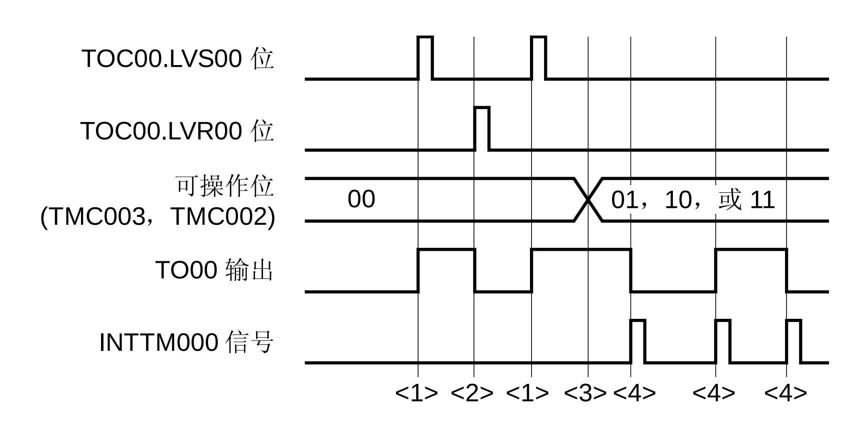
可以按照下列过程设置 LVS00 和 LVR00。

图 6-57. 设置 LVS00 和 LVR00 位的流程示例



注意事项 **请确保按照上述的<1>、<2>和<3>步骤设置 LVS00 和 LVR00。**
步骤<2>可以在<1>之后和<3>之前执行。

图 6-58. LVR00 和 LVS00 时序示例



<1> 当 LVS00 与 LVR00 = 10 时，TO00 输出变为高电平。

<2> 当 LVS00 与 LVR00 = 01 时，TO00 输出变为低电平（即使 LVS00 与 LVR00 被清除为 00，引脚输出保持高电平不变）。

<3> 当 TMC003 与 TMC002 被设置为 01、10 或 11 时，定时器开始操作。因为在开始操作前 LVS00 与 LVR00 被设置为 10，所以 TO00 输出从高电平开始。定时器开始操作之后，禁止设置 LVS00 与 LVR00，直到 TMC003 与 TMC002 = 00（禁止定时器操作）。

<4> 每当产生中断信号（INTTM000）时，反转 TO00 的输出电平。

6.6 16 位定时器/事件计数器 00 注意事项

（1） 16 位定时器/事件计数器 00 各个通道的限制条件

表 6-3 显示了各个通道的限制条件。

表 6-3. 16 位定时器/事件计数器 00 各个通道的限制条件

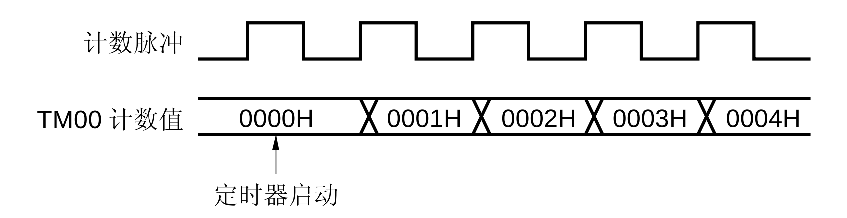
操作	限制条件
用作间隔定时器	–
用作方波输出	
用作外部事件计数器	
用于通过输入 TI000 引脚的有效边沿进入清零&启动模式	当用来检测 TI010 引脚的有效边沿时，禁止使用定时器输出（TO00）。（TOC00 = 00H）
用作自由运行定时器	–
用作 PPG 输出	0000H ≤ CR010 < CR000 ≤ FFFFH
用作单次触发脉冲输出	禁止设置 CR000 和 CP010 为相同的值。
用作脉冲宽度测量	禁止使用定时器输出（TO00）（TOC00 = 00H）

（2） 定时器启动误差

定时器启动后，在产生匹配信号时可能产生最大一个时钟周期的误差。这是因为 TM00 计数操作的启动和计数脉冲不同步引起的。



图 6-59. TM00 计数启动时序



（3） CR000 与 CR010 的设置（TM00 与 CR000 匹配时进入清零&启动模式）

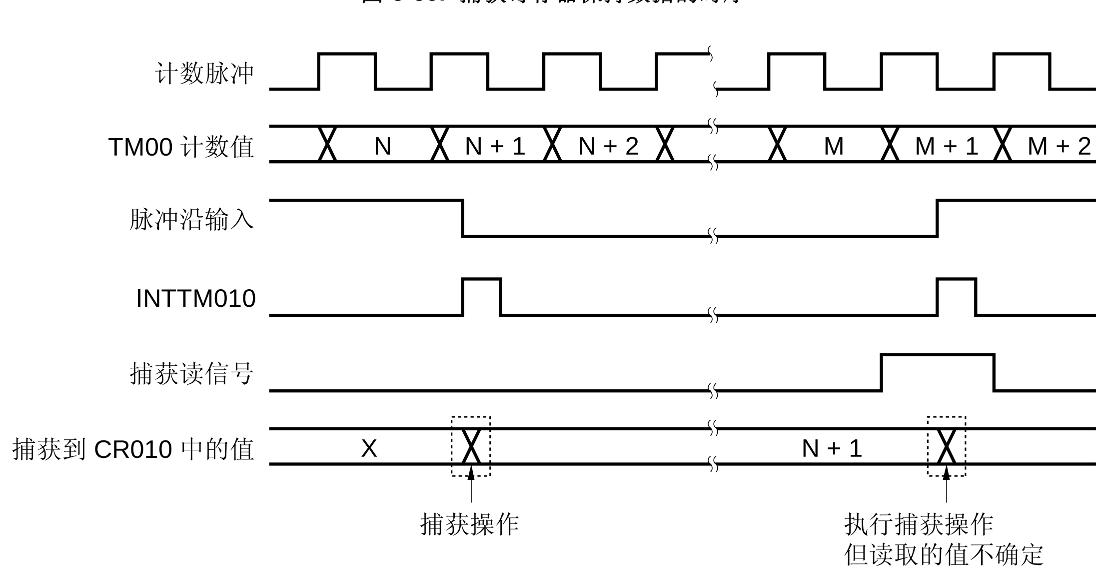
将 CR000 与 CR010 设置为 0000H 以外的值（当 TM00 用作外部事件计数器时，不能计数一个脉冲）。

（4） 捕获寄存器保持数据的时序

（a）如果在读取 CR000/CR010 时，有效边沿输入到 TI000/TI010 引脚并且检测到 TI000 引脚的反向沿时，CR010 执行捕获操作，但不保证 CR000/CR010 的读取值。此时，当检测到 TI000/TI010 引脚的有效边沿时产生中断信号（INTTM000/INTTM010）（当检测到 TI000 引脚的反向沿时不产生中断信号）。

如果因为检测到 TI000/TI010 引脚的有效边沿而捕获计数值时，则应在 INTTM000/INTTM010 信号产生后读取 CR000/CR010 的值。

图 6-60. 捕获寄存器保持数据的时序



（b） 在 16 位定时器/事件计数器 00 停止操作后，CR000 与 CR010 的值不保证。

（5） 设置有效边沿

当定时器停止操作时（TMC003 与 TMC002 = 00）设置 TI000 引脚的有效边沿。使用 ES000 与 ES001 设置有效边沿。

（6） 单次触发脉冲的重新触发

必须确定在单次触发脉冲输出模式下正在输出活动电平时，没有触发产生。请确保在当前活动电平输出之后，输入下一个触发。

（7） OVF00 标志的操作

（a） 设置 OVF00 标志（1）

在以下情况下或者当 TM00 溢出时，OVF00 标志被设置为 1。

选择在 TM00 与 CR000 匹配时进入清零&启动模式。

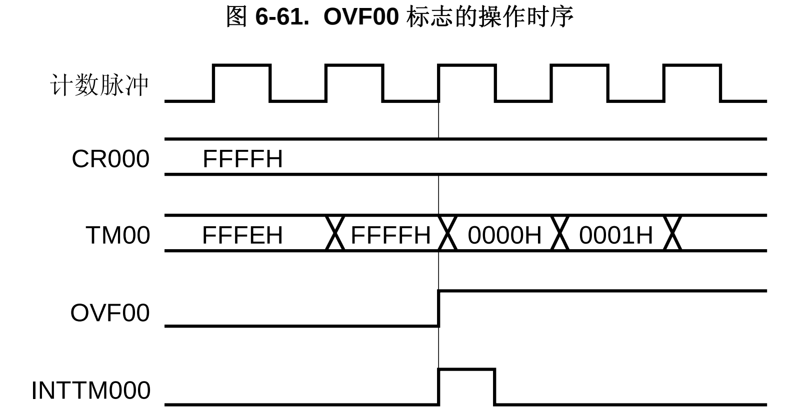
↓

设置 CR000 为 FFFFH。

↓

当 TM00 与 CR000 匹配且 TM00 从 FFFFH 变为 0000H 时，TM00 被清除。

图 6-61. OVF00 标志的操作时序



（b） OVF00 标志清零

即使在 TM00 溢出后对下一个计数时钟计数之前（TM00 的值变为 0001H 之前）将 OVF00 标志清除为 0，它会被再次设置为 1，并且对其清除是无效的。

（8） 单次触发脉冲输出

在自由运行定时器模式下或者在通过输入 TI000 引脚的有效边沿进入清零&启动模式下，单次触发脉冲输出可以正确操作。而在 TM00 与 CR000 匹配时进入清零&启动模式下，不能输出单次触发脉冲。

（9） 捕获操作

（a） 当 TI000 的有效边沿被指定作为计数时钟时

当 TI000 的有效边沿被指定作为计数时钟时，指定 TI000 作为触发的捕获寄存器不能正确操作。

（b） 通过 TI010 与 TI000 引脚输入信号准确捕获计数值的脉冲宽度

为了准确捕获计数值，作为捕获触发的 TI000 与 TI010 引脚的输入脉冲宽度必须大于由 PRM00 选择的计数时钟两个周期（参见图 **6-7**）。

（c） 中断信号的产生

在计数时钟的下降沿执行捕获操作，但在下一个计数时钟的上升沿处产生中断信号（INTTM000 与 INTTM010）（参见 图 **6-7**）。

（d） 当 CRC001（捕获/比较控制寄存器 00（CRC00）的第 1 位）被设置为 1 时的注意事项

在输入 TI000 引脚的信号的反向沿将 TM00 寄存器的计数值捕获到 CR000 寄存器时，捕获计数值之后不产生中断信号（INTTM000）。如果在该操作期间，检测到有效边沿输入 TI010 引脚，则不执行捕获操作，但会产生中断信号（INTTM000）作为外部中断信号。不使用外部中断时，屏蔽 INTTM000 信号。

（10） 边沿检测

（a） 复位后指定有效边沿

如果复位后在 TI000 或 TI010 引脚为高电平且指定 TI000 或 TI010 引脚的有效边沿为上升沿或双沿时，使能 16 位定时器/事件计数器 00 的操作，则 TI000 或 TI010 引脚的高电平将被检测认为是上升沿。当 TI000 或 TI010 引脚被上拉时要注意。但是，一旦定时器操作已经被停止然后再次使能时，不检测上升沿。

（b） 用于消除噪声的采样时钟

根据 TI000 的有效边沿是用作计数时钟还是用作捕获触发，用于消除噪声的采样时钟会发生变化。在前一种情况下采样时钟固定为 f_{PRS}。而在后一种情况下，则使用 PRM00 选择的计数时钟进行采样。

当输入 TI000 引脚的信号被采样且连续两次检测到有效电平时，才认为检测到有效边沿。因此，可以消除窄脉冲宽度的噪声（参见 图 **6-7**）。

（11） 定时器操作

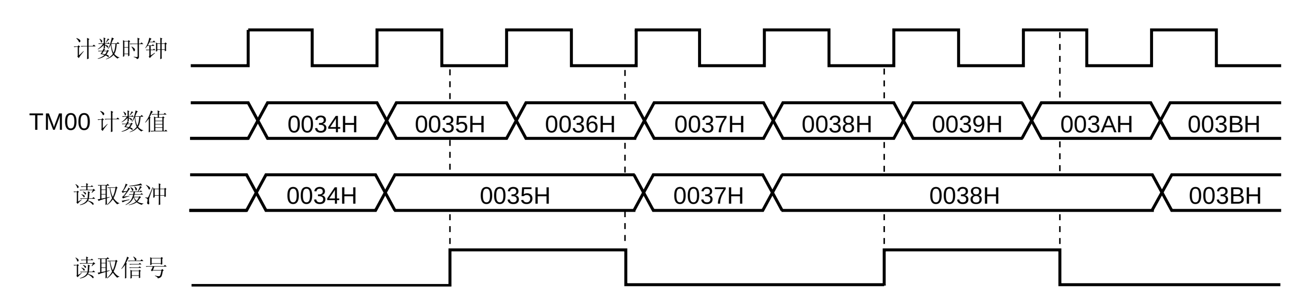
无论 CPU 处于何种操作模式，当定时器停止操作时，不响应 TI000/TI010 引脚的输入信号。

备注 f_{PRS}： 外设硬件时钟频率

<R> **（12） 16 位定时器计数器 00（TM00）的读取**

无需停止实际的计数器就可以读取 TM00，因为读取已经被捕获到缓冲器中的计数值时，该值是固定的。但是，在计数起向上计数之前的瞬间读取时，该缓冲器可能还没有被更新，因为该缓冲器是在计数器向上计数的时序时被更新。

图 6-62. 16 位定时器计数器 00（TM00）读取时序



第七章 8 位定时器/事件计数器 50、51 和 52

7.1 8 位定时器/事件计数器 50、51 和 52 的功能

8 位定时器/事件计数器 50、51 和 52 具有以下功能

- 间隔定时器
- 外部事件计数器注 1
- 方波输出注 2
- PWM 输出注 2

注　**1.** TM52 和 TM00 可以级联用作外部 24 位事件计数器。同样，通过 TMH2 控制 TM52 的外部事件输入的输入使能。详情参见 **第六章 16 位定时器/事件计数器 00**。

　　2. 仅限 TM50 和 TM51 。

7.2 8 位定时器/事件计数器 50、51 和 52 的配置

8 位定时器/事件计数器 50、51 和 52 包括以下硬件。

表 7-1. 8 位定时器/事件计数器 50、51 和 52 的配置

项目	配置
定时器寄存器	8 位定时器计数器 5n （TM5n）
寄存器	8 位定时器比较寄存器 5n （CR5n）
定时器输入	TI5n
定时器输出	TO50，TO51
控制寄存器	定时器时钟选择寄存器 5n （TCL5n） 8 位定时器模式控制寄存器 5n （TMC5n） 输入切换控制寄存器（ISC） 端口模式寄存器 3（PM3）或端口模式寄存器 4（PM4） 端口寄存器 3（P3）或端口寄存器 4（P4）

备注　n = 0 至 2

图 7-1 至 7-3 显示了 8 位定时器/事件计数器 50、51 和 52 的框图。

图 7-1. 8 位定时器/事件计数器 50 的框图

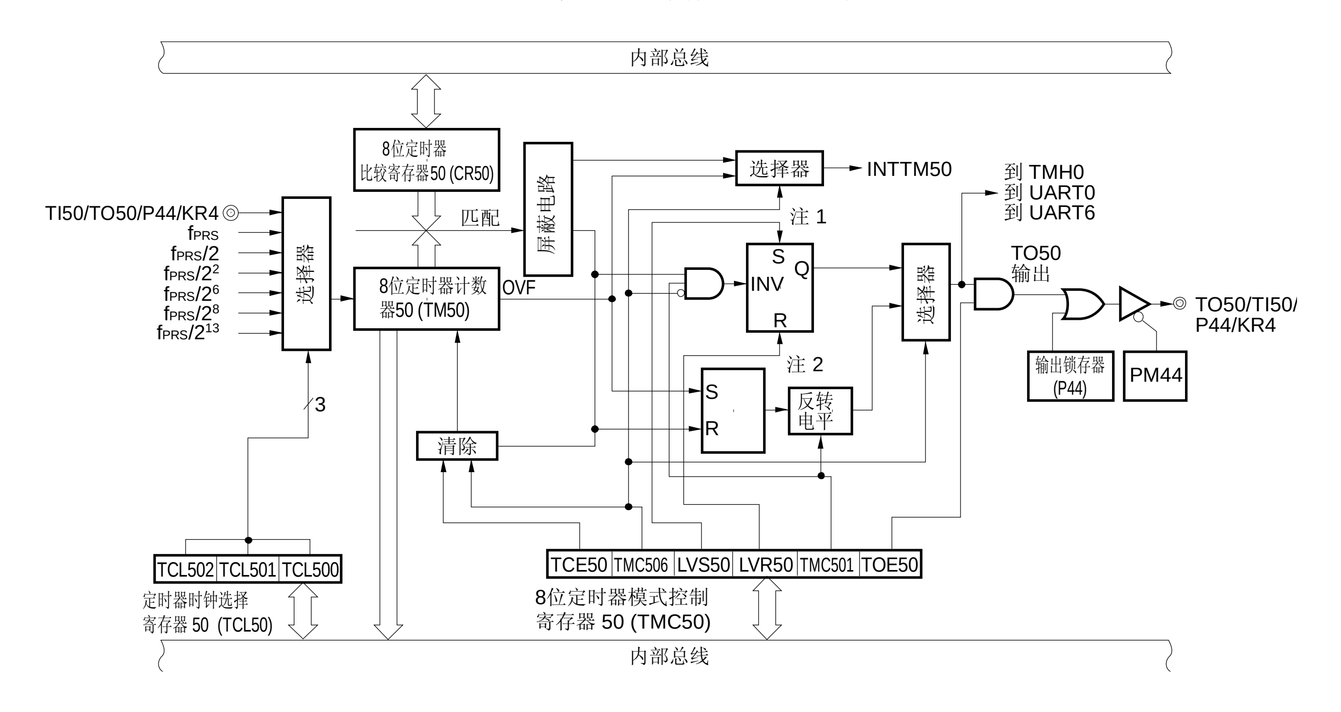
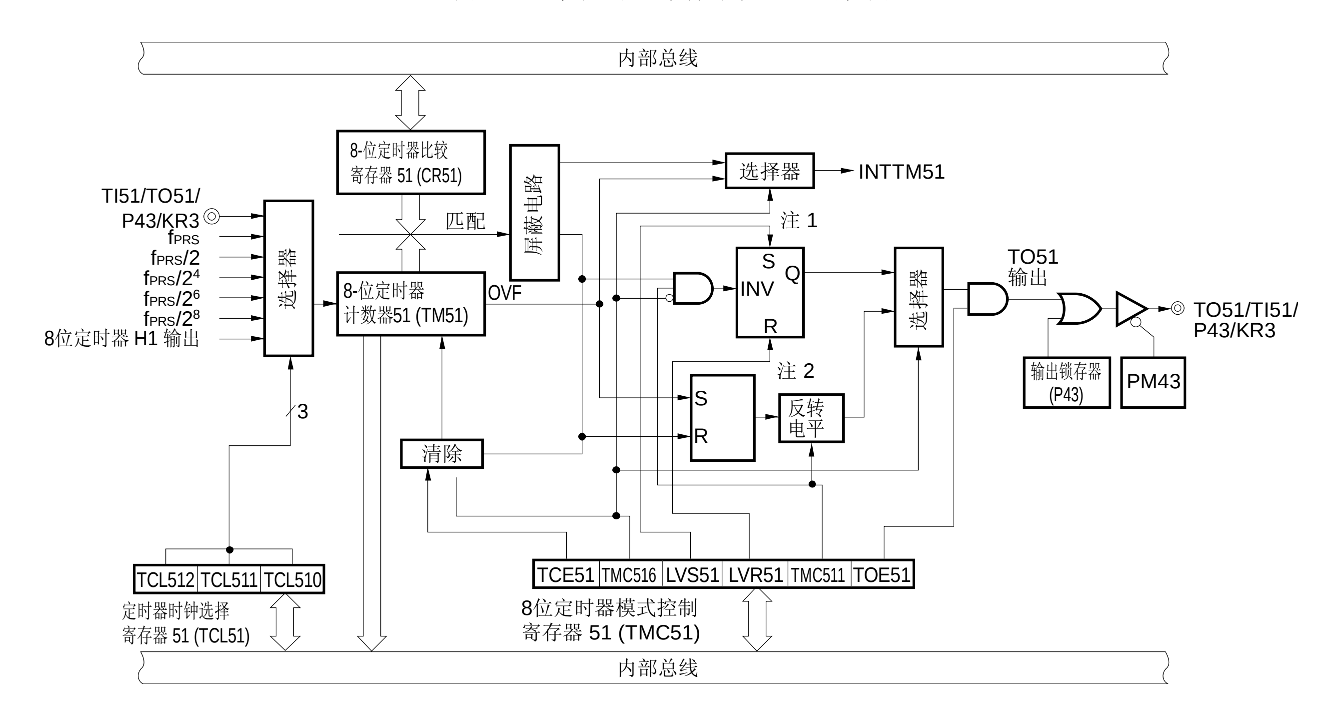


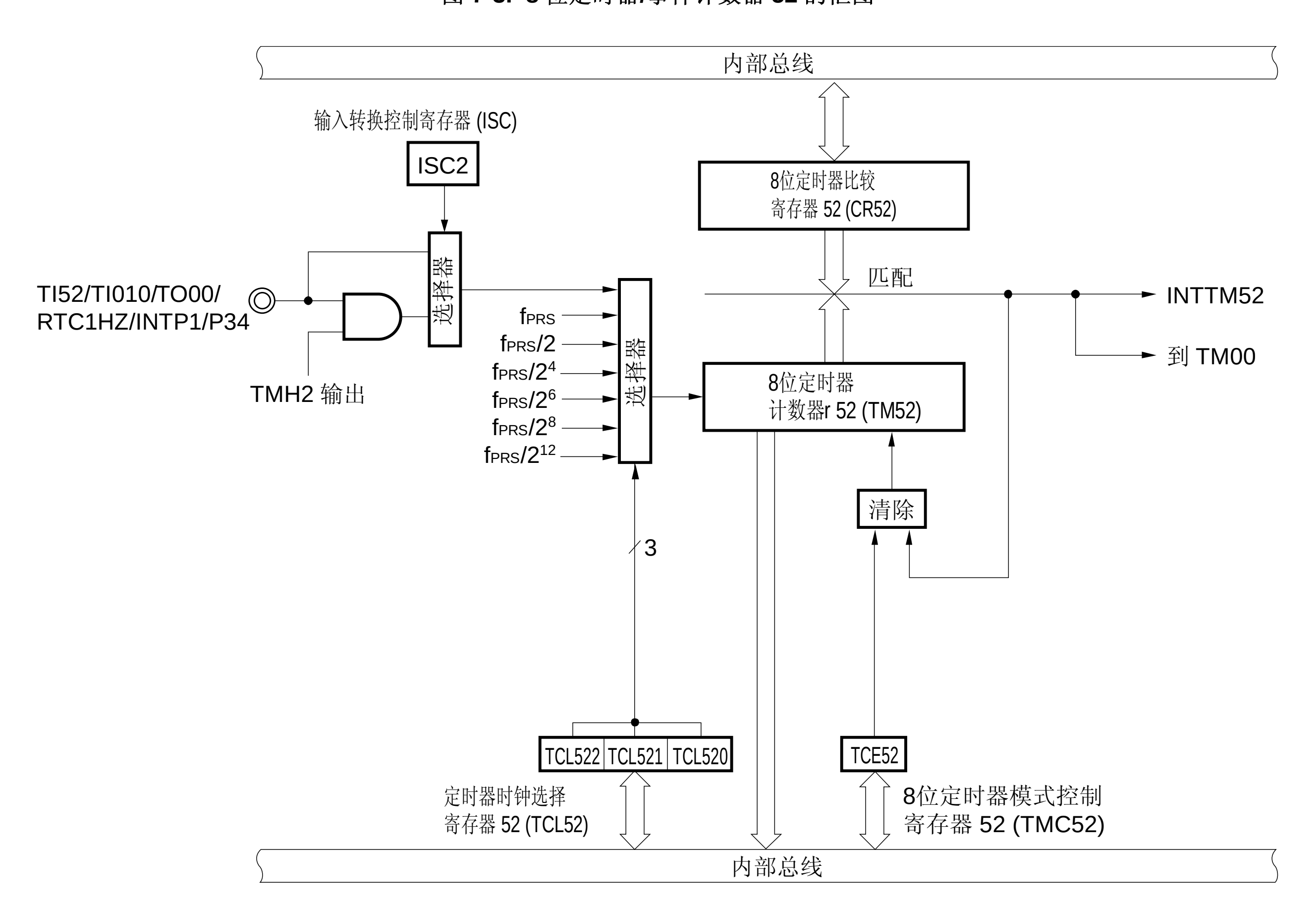
图 7-2. 8 位定时器/事件计数器 51 的框图



注 **1.** 定时器输出 F/F

2. PWM 输出 F/F

图 7-3. 8 位定时器/事件计数器 52 的框图



（1）　8 位定时器计数器 5n（TM5n）

TM5n 是 8 位寄存器，用于对计数脉冲进行计数，并且是只读的。

该计数器的计数值随计数时钟的上升沿同步增加。

图 7-4. 8 位定时器计数器 5n（TM5n）的格式

地址: FF16H (TM50), FF6FH (TM51), FF51H (TM52)　　复位后: 00H　　R

符号	7	6	5	4	3	2	1	0
TM5n (n = 0-2)								

在以下几种情况下将计数值清除为 00H。

<1>　产生复位信号

<2>　TCE5n 清零

<3>　在 TM5n 与 CR5n 匹配时进入清零&启动的模式下，当 TM5n 与 CR5n 匹配时。

（2）　8 位定时器比较寄存器 5n（CR5n）

可以通过 8 位存储器操作指令对 CR5n 进行读取和写入。

除 PWM 模式外，CR5n 的设置值将与 8 位定时器计数器 5n（TM5n）的计数值进行比较，如果匹配，则产生中断请求（INTTM5n）。

在 PWM 模式下，TM5n 与 CR5n 的值匹配时，TO5n 输出变为无效，但不产生中断信号。

CR5n 可以设置的值的范围是 00H 至 FFH。

复位信号的产生会将 CR5n 清除为 00H。

图 7-5. 8 位定时器比较寄存器 5n（CR5n）的格式

地址: FF17H (CR50), FF41H (CR51), FF59H (CR52)　　复位后: 00H　　R/W

符号	7	6	5	4	3	2	1	0
CR5n (n = 0-2)								

注意事项　**1.　在 TM5n 与 CR5n 匹配时进入清零&启动（TMC5n6 = 0）的模式下，在操作期间不要将其它值写入 CR5n。**

2.　在 PWM 模式下，重写 CR5n 的间隔至少要大于计数时钟（由 TCL5n 选择时钟）的 3 个周期。

备注　　n = 0 至 2

7.3 控制 8 位定时器/事件计数器 50、51 和 52 的寄存器

以下五种寄存器用于控制 8 位定时器/事件计数器 50、51 和 52。

- 定时器时钟选择寄存器 5n （TCL5n）
- 8 位定时器模式控制寄存器 5n （TMC5n）
- 输入切换控制寄存器 （ISC）
- 端口模式寄存器 3（PM3）或端口模式寄存器 4（PM4）
- 端口寄存器 3（P3）或端口寄存器 4（P4）

（1） 定时器时钟选择寄存器 5n （TCL5n）

该寄存器用于设置 8 位定时器/事件计数器 5n 的计数时钟，和 TI5n 引脚输入的有效边沿。
可以由 1 位或 8 位存储器操作指令来设置 TCL5n。
复位信号的产生会将 TCL5n 清除为 00H。

备注 n = 0 至 2

图 7-6. 定时器时钟选择寄存器 50（TCL50）的格式

地址： FF6AH 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
TCL50	0	0	0	0	0	TCL502	TCL501	TCL500

TCL502	TCL501	TCL500	计数时钟选择[注 1]			
				f_{PRS} = 2 MHz	f_{PRS} = 5 MHz	f_{PRS} = 10 MHz
0	0	0	TI50 引脚下降沿			
0	0	1	TI50 引脚上升沿			
0	1	0	f_{PRS}[注 2]	2 MHz	5 MHz	10 MHz
0	1	1	$f_{PRS}/2$	1 MHz	2.5 MHz	5 MHz
1	0	0	$f_{PRS}/2^2$	500 kHz	1.25 MHz	2.5 MHz
1	0	1	$f_{PRS}/2^6$	31.25 kHz	78.13 kHz	156.25 kHz
1	1	0	$f_{PRS}/2^8$	7.81 kHz	19.53 kHz	39.06 kHz
1	1	1	$f_{PRS}/2^{13}$	0.24 kHz	0.61 kHz	1.22 kHz

注 1. 如果外设硬件时钟（f_{PRS}）使用高速系统时钟（f_{XH}）（XSEL ＝ 1），f_{PRS} 操作频率会根据供电电压不同而变动。
- V_{DD} = 2.7 至 5.5 V： $f_{PRS} \leq 10$ MHz
- V_{DD} = 1.8 至 2.7 V： $f_{PRS} \leq 5$ MHz

2. 如果外设硬件时钟（f_{PRS}） 使用内部高速振荡时钟（f_{XH}）（XSEL = 0），当 $1.8 V \leq V_{DD} < 2.7 V$ 时，禁止设置 TCL502，TCL501，TCL500 = 0，1，0 （计数时钟：f_{PRS}） 。

注意事项 1. 如果要向 TCL50 写入其它值，则必须先停止定时器操作。
2. 请确保将第 3 位至第 7 位清除为 0。

备注 f_{PRS}： 外设硬件时钟频率

图 7-7. 定时器时钟选择寄存器 51（TCL51）的格式

地址： FF8CH　复位后： 00H　R/W

符号	7	6	5	4	3	2	1	0
TCL51	0	0	0	0	0	TCL512	TCL511	TCL510

TCL512	TCL511	TCL510	计数时钟选择注 1			
				f_{PRS} = 2 MHz	f_{PRS} = 5 MHz	f_{PRS} = 10 MHz
0	0	0	TI51 引脚下降沿			
0	0	1	TI51 引脚上升沿			
0	1	0	f_{PRS} 注 2	2 MHz	5 MHz	10 MHz
0	1	1	$f_{PRS}/2$	1 MHz	2.5 MHz	5 MHz
1	0	0	$f_{PRS}/2^4$	125 kHz	312.5 kHz	625 kHz
1	0	1	$f_{PRS}/2^6$	31.25 kHz	78.13 kHz	156.25 kHz
1	1	0	$f_{PRS}/2^8$	7.81 kHz	19.53 kHz	39.06 kHz
1	1	1	定时器 H1 输出信号			

注　**1.** 如果外设硬件时钟（f_{PRS}）使用高速系统时钟（f_{XH}）（XSEL = 1），f_{PRS} 操作频率会根据供电电压不同而变动。

- V_{DD} = 2.7 至 5.5 V： $f_{PRS} \leq 10$ MHz
- V_{DD} = 1.8 至 2.7 V： $f_{PRS} \leq 5$ MHz

2. 如果外设硬件时钟（f_{PRS}） 使用内部高速振荡时钟（f_{XH}）（XSEL = 0），当 1.8 V ≤ V_{DD} < 2.7 V 时，禁止设置 TCL512，TCL511，TCL510 = 0，1，0 （计数时钟：f_{PRS}） 。

注意事项　1. 如果要向 TCL51 写入其它值，则必须先停止定时器操作。

2. 请确保将第 3 位至第 7 位清除为 0。

图 7-8. 定时器时钟选择寄存器 52 （TCL52） 的格式

地址： FF5BH 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
TCL52	0	0	0	0	0	TCL522	TCL521	TCL520

TCL522	TCL521	TCL520	计数时钟选择[注 1]			
				f_{PRS} = 2 MHz	f_{PRS} = 5 MHz	f_{PRS} = 10 MHz
0	0	0	由 ISC2 选择时钟的下降沿			
0	0	1	由 ISC2 选择时钟的上升沿			
0	1	0	f_{PRS}[Note 2]	2 MHz	5 MHz	10 MHz
0	1	1	$f_{PRS}/2$	1 MHz	2.5 MHz	5 MHz
1	0	0	$f_{PRS}/2^4$	125 kHz	312.5 kHz	625 kHz
1	0	1	$f_{PRS}/2^6$	31.25 kHz	78.13 kHz	156.25 kHz
1	1	0	$f_{PRS}/2^8$	7.81 kHz	19.53 kHz	39.06 kHz
1	1	1	$f_{PRS}/2^{12}$	0.49 kHz	1.22 kHz	2.44 kHz

注 1. 如果外设硬件时钟（f_{PRS}）使用高速系统时钟（f_{XH}）（XSEL = 1），f_{PRS} 操作频率会根据供电电压不同而变动。

- V_{DD} = 2.7 至 5.5 V： $f_{PRS} \leq 10$ MHz
- V_{DD} = 1.8 至 2.7 V： $f_{PRS} \leq 5$ MHz

2. 如果外设硬件时钟（f_{PRS}） 使用内部高速振荡时钟（f_{XH}）（XSEL = 0），当 1.8 V ≤ V_{DD} < 2.7 V 时，禁止设置 TCL522，TCL521，TCL520 = 0，1，0（计数时钟：f_{PRS}） 。

注意事项 1. 如果要向 TCL52 写入其它值，则必须先停止定时器操作。

2. 请确保将第 3 位至第 7 位清除为 0。

备注 f_{PRS}： 外设硬件时钟频率

（2）　8 位定时器模式控制寄存器 5n （TMC5n）

TMC5n 寄存器用于执行以下五种类型的设置。

<1>　8 位定时器计数器 5n（TM5n）计数操作控制
<2>　8 位定时器计数器 5n（TM5n）操作模式选择[注]
<3>　定时器输出 F/F（双稳态触发器）状态设置[注]
<4>　定时器 F/F 控制或 PWM（自由运行）模式下有效电平的选择[注]
<5>　定时器输出控制[注]

可以由 1 位或 8 位存储器操作指令来设置 TMC5n。
复位信号的产生会将该寄存器清除为 00H。

注　　仅限 TM50 和 TM51 。
备注　　n = 0 至 2

图 7-9. 8 位定时器模式控制寄存器 50（TMC50）的格式

地址： FF6BH　复位后： 00H　R/W [注]

符号	<7>	6	5	4	<3>	<2>	1	<0>
TMC50	TCE50	TMC506	0	0	LVS50	LVR50	TMC501	TOE50

TCE50	TM50 计数操作控制
0	清除为 0 后，禁止计数操作 （计数器停止操作）
1	开始计数

TMC506	TM50 操作模式选择
0	清零&启动模式发生在 TM50 与 CR50 匹配时
1	PWM （自由运行）模式

LVS50	LVR50	定时器输出 F/F 状态设置
0	0	无变化
0	1	定时器输出 F/F 清除（0） （TO50 输出的默认值：低电平）
1	0	定时器输出 F/F 置位（1） （TO50 输出的默认值：高电平）
1	1	禁止设置

TMC501	其他模式（TMC506 = 0）	PWM 模式（TMC506 = 1）
	定时器 F/F 控制	有效电平选择
0	禁止反转操作	高电平有效
1	使能反转操作	低电平有效

TOE50	定时器输出控制
0	禁止输出（TO50 输出为低电平）
1	使能输出

注　　第 2 位和第 3 位只能写入。

（**注意事项**和**备注**在下页列出）

图 7-10. 8 位定时器模式控制寄存器 51 （TMC51） 的格式

地址： FF43H 复位后： 00H R/W[注]

符号	<7>	6	5	4	<3>	<2>	1	<0>
TMC51	TCE51	TMC516	0	0	LVS51	LVR51	TMC511	TOE51

TCE51	TM51 计数操作控制
0	清除为 0 后，禁止计数操作 （计数器停止操作）
1	开始计数

TMC516	TM51 操作模式选择
0	清零&启动模式发生在 TM51 与 CR51 匹配时
1	PWM （自由运行）模式

LVS51	LVR51	定时器输出 F/F 状态设置
0	0	无变化
0	1	定时器输出 F/F 清除（0）（TO51 输出的默认值：低电平）
1	0	定时器输出 F/F 置位（1）（TO51 输出的默认值：高电平）
1	1	禁止设置

TMC511	其他模式 （TMC516 = 0）	PWM 模式 （TMC516 = 1）
	定时器 F/F 控制	有效电平选择
0	禁止反转操作	高电平有效
1	使能反转操作	低电平有效

TOE51	定时器输出控制
0	禁止输出 （TO51 输出为低电平）
1	使能输出

注 第 2 位和第 3 位只能写入。

注意事项 1.除 PWM 模式外，LVS5n 与 LVR5n 的设置都有效。

2.按下列顺序执行<1>至<4>步骤，注意不能同时执行。

<1>设置 TMC5n1，TMC5n6： 操作模式设置

<2>设置 TOE5n 为使能输出： 定时器输出使能

<3>设置 LVS5n，LVR5n（参见注意事项 1）： 定时器 F/F 设置

<4>设置 TCE5n

3.当 TCE5n = 1 时，禁止设置 TMC5n 的其他位。

4.实际上除 TO5n 输出，TO50/TI50/P44/KR4 和 TO51/TI51/P43/KR3 引脚输出由 PM44 和 P44、PM43 和 P43 决定。

备注 **1.** 在 PWM 模式下，将 TCE5n 清除为 0 可以使 PWM 输出无效。

2. 读取 LVS5n 与 LVR5n 时，读取的值为 0。

3.无论 TCE5n 的值如何，TMC5n6、LVS5n、LVR5n、TMC5n1 和 TOE5n 的值都在 TO5n 引脚上反映。

4. n = 0，1

图 7-11. 8 位定时器模式控制寄存器 52 （TMC52）的格式

地址： FF5CH 复位后： 00H R/W

符号	<7>	6	5	4	3	2	1	0
TMC52	TCE52	0	0	0	0	0	0	0

TCE52	TM52 计数操作控制
0	清除为 0 后，禁止计数操作 （计数器停止操作）
1	开始计数

注意事项 请确保将第 0 位至第 6 位清除为 0。

（3） 输入切换控制寄存器（ISC）

将 ISC2 置为 1，可以通过 TOH2 输出信号来控制 TI52 输入信号。

可以由 1 位或 8 位存储器操作指令来设置该寄存器。

复位信号的产生会将该寄存器清除为 00H。

图 7-12. 输入切换控制寄存器（ISC）的格式

地址： FF4FH 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
ISC	0	0	ISC5	ISC4	ISC3	ISC2	ISC1	ISC0

ISC5	ISC4	TxD6，RxD6 输入源的选择
0	0	TxD6：P112，RxD6： P113
0	1	TxD6：P13，RxD6： P12
其他情况		禁止设置

ISC3	RxD6/P113 输入使能/禁止
0	禁止 RxD6/P113 输入
1	使能 RxD6/P113 输入

ISC2	TI52 输入源的控制
0	不使能对 TI52 输入的控制（P34）
1	使能对 TI52 输入的控制（P34）[注 1]

ISC1	选择 TI000 输入源
0	TI000 （P33）
1	RxD6 （P12 或 P113[注 2]）

ISC0	选择 INTP0 输入源
0	INTP0 （P120）
1	RxD6 （P12 或 P113[注 2]）

注 **1.** TI52 输入由 TOH2 输出信号来控制。

2. 由 ISC5 和 ISC4 选择 P12 或 P113。

（4） 端口寄存器 3 和 4 （PM3，PM4）

这些寄存器可以按位设置端口 3 和 4 的输入/输出模式。

当 P44/TO50/TI50/KR4 和 P43/TO51/TI51/KR3 引脚用作定时器输出时，清除 PM44 和 PM43 为 0，将 P44 和 P43 的输出锁存清除为 0。

当 P44/TO50/TI50/KR4、P43/TO51/TI51/KR3 和 P34/TI52/TI010/TO00/RTC1HZ/INTP1 引脚用作定时器输入时，将 PM44，PM43 和 PM34 置为 1。此时 P44 的输出锁存、PM43 和 PM34 可能为 0 或为 1。

可以由 1 位或 8 位存储器操作指令来设置 PM3 和 PM4 。

复位信号的产生会将寄存器的内容设置为 FFH。

图 7-13. 端口模式寄存器 3（PM3）的格式

地址： FF23H 复位后： FFH R/W

符号	7	6	5	4	3	2	1	0
PM3	1	1	1	PM34	PM33	PM32	PM31	0

PM3n	P3n 引脚 I/O 模式选择 （n = 1 至 4）
0	输出模式 （输出缓冲器打开）
1	输入模式 （输出缓冲器关闭）

图 7-14. 端口模式寄存器 4（PM4）的格式

地址： FF24H 复位后： FFH R/W

符号	7	6	5	4	3	2	1	0
PM4	1	1	1	PM44	PM43	PM42	PM41	PM40

PM4n	P4n 引脚 I/O 模式选择 （n = 0 至 4）
0	输出模式（输出缓冲器打开）
1	输入模式（输出缓冲器关闭）

7.4 8 位定时器/事件计数器 50、51 和 52 的操作

7.4.1 用作间隔定时器

8 位定时器/事件计数器 5n 用作间隔定时器，可以按照 8 位定时器比较寄存器 5n（CR5n）内预置的计数值间隔时间重复产生中断请求。

当 8 位定时器计数器 5n（TM5n）的计数值与 CR5n 的设置值匹配时，将 TM5n 的值清除为 0 并继续计数，同时产生中断请求信号（INTTM5n）。

根据定时器时钟选择寄存器 5n（TCL5n）的第 0 位至第 2 位（TCL5n0 至 TCL5n2），选择 TM5n 的计数时钟。

设置

<1> 设置寄存器
- TCL5n： 设置计数时钟
- CR5n： 比较值
- TMC5n： 停止计数操作，选择 TM5n 与 CR5n 匹配时发生清零&启动的模式。
  （TMC5n = 0000×××0B， × = 无须理会）

<2> 在设置 TCE5n=1 之后，计数操作开始。

<3> 如果 TM5n 的值与 CR5n 值匹配，产生中断 INTTM5n （将 TM5n 清除为 00H）。

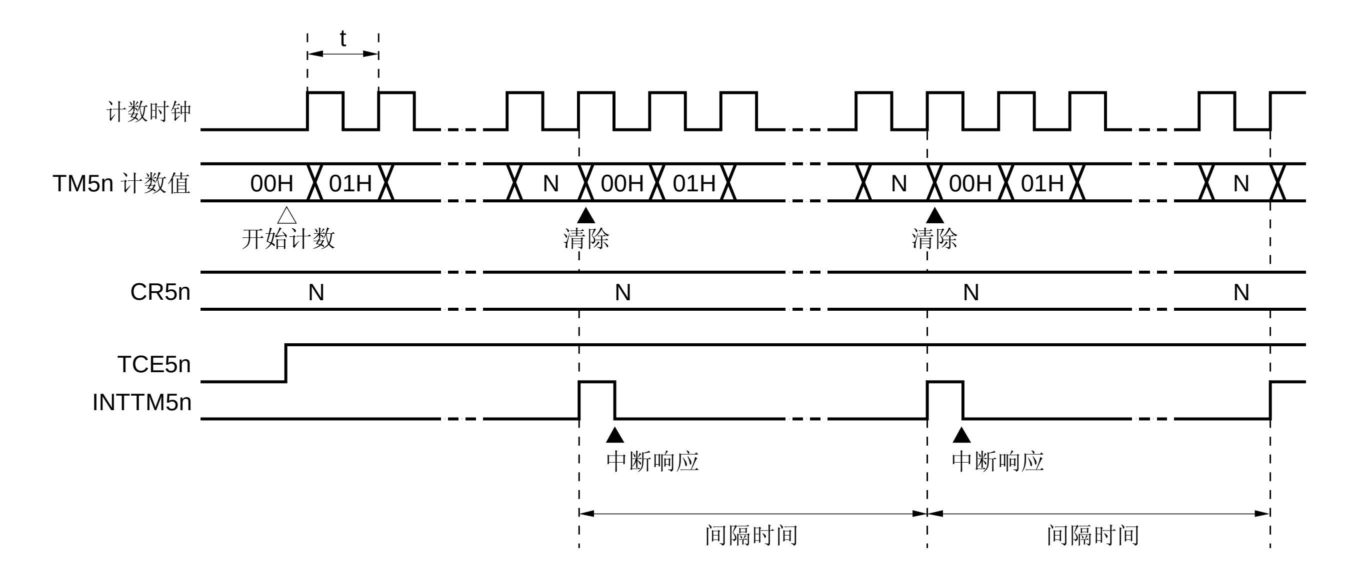
<4> 按照相同的时间间隔重复产生 INTTM5n。
将 TCE5n 设置为 0，可以停止计数操作。

注意事项 操作期间不要向 CR5n 写入其它值。

备注 **1.** 如需了解如何使能 INTTM5n 中断，参见 **第二十章 中断功能。**
2. n = 0 至 2

图 7-15. 间隔定时器的操作时序（1/2）

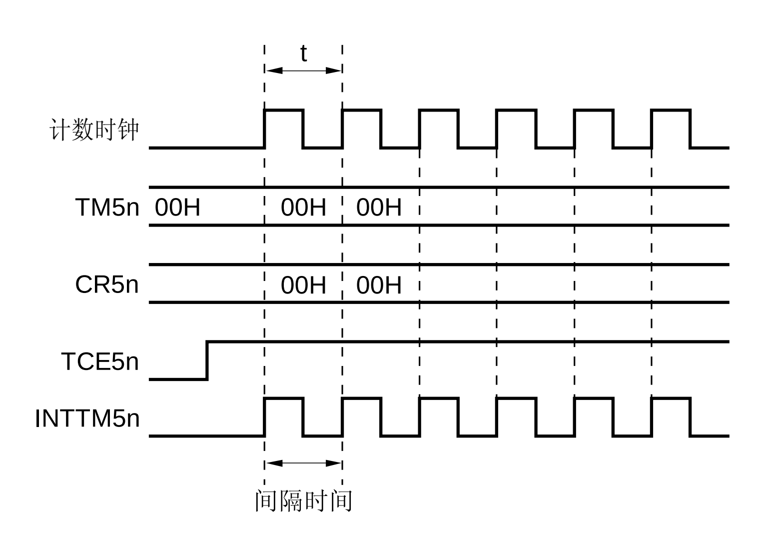
（a） 基本操作



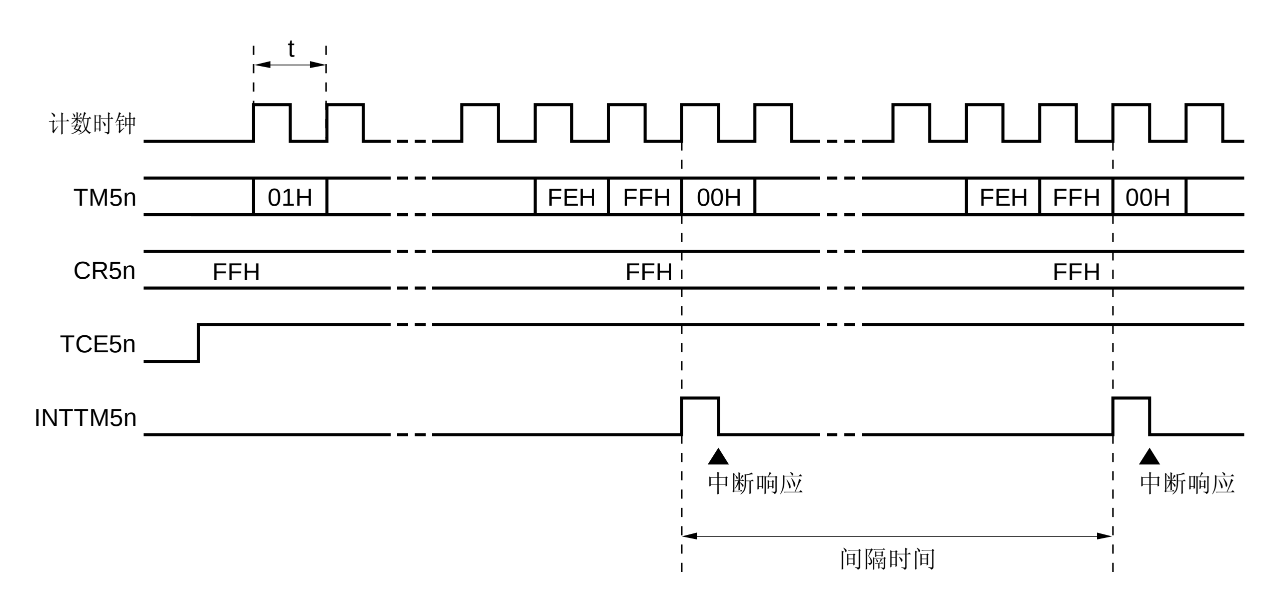
备注 间隔时间= （N + 1） × t
N = 01H 至 FFH
n = 0 至 2

图 7-15. 间隔定时器的操作时序（2/2）

（b）　当 CR5n = 00H



（c）　当 CR5n = FFH



备注　　n = 0 至 2

7.4.2 用作外部事件计数器

外部事件计数器通过 8 位定时器计数器 5n（TM5n）对输入到 TI5n 引脚的外部时钟脉冲数量计数。

每次当有效边沿输入时，TM5n 递增。有效边沿由定时器时钟选择寄存器 5n（TCL5n）指定，可以选择上升沿或下降沿。

当 TM5n 的计数值与 8 位定时器比较寄存器 5n（CR5n）的值匹配时，将 TM5n 清除为 0，并产生中断请求信号（INTTM5n）。

只要 TM5n 的值与 CR5n 的值匹配，就产生 INTTM5n。

设置

<1> 设置各个寄存器

- 设置端口模式寄存器（PM44，PM43 或 PM34）[注] 为 1。
- TCL5n：　选择 TI5n 引脚输入边沿
  TI5n 引脚下降沿 → TCL5n = 00H
  TI5n 引脚上升沿 → TCL5n = 01H
- CR5n：　比较值
- TMC5n：　停止计数，选择 TM5n 与 CR5n 匹配时发生清零&启动的模式，禁止定时器 F/F 反转操作，禁止定时器输出。
  （TMC5n = 0000××00B　× =无须理会）

<2> 当设置 TCE5n = 1 时，开始对从 TI5n 引脚输入的脉冲数量计数。

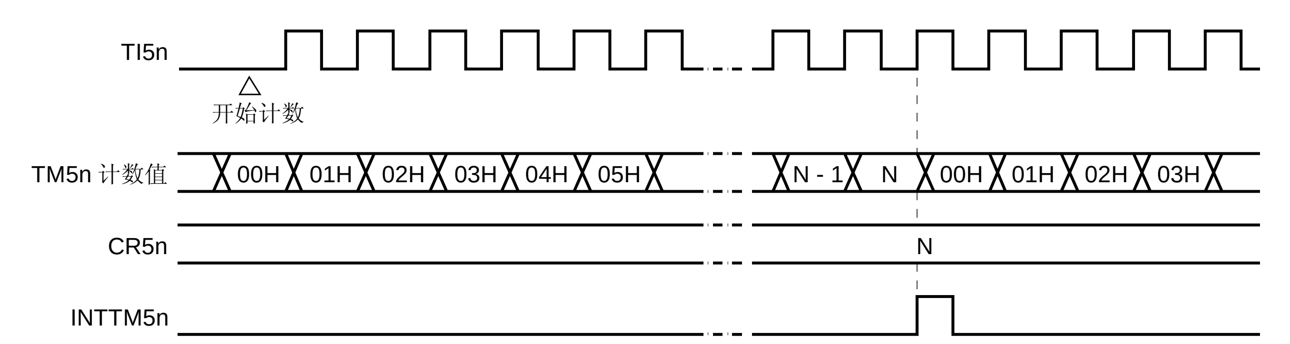
<3> 当 TM5n 的值与 CR5n 匹配时，将产生 INTTM5n（TM5n 被清除为 00H）。

<4> 在经过以上设置后，每当 TM5n 与 CR5n 的值匹配时，都会产生 INTTM5n。

注　8 位定时器/事件计数器 50：　PM44
　　8 位定时器/事件计数器 51：　PM43
　　8 位定时器/事件计数器 52：　PM34

备注　如需了解如何使能 INTTM5n 中断，参见 **第二十章　中断功能**。

图 7-16.　外部事件计数器的操作时序（指定上升沿有效）



备注　**1.**　将 8 位定时器/事件计数器 52（TM52）和 16 位定时器/事件计数器（TM00）级联，可以用作外部 24 位事件计数器。TM52 的输入使能由 TMH2 控制。详情参见 **6.4.9　外部 24 位事件计数器操作**。

2.　N = 00H 至 FFH，n = 0 至 2

7.4.3 方波输出操作

按照 8 位定时器比较寄存器 5n（CR5n）的预置值决定的间隔时间，输出任选频率的方波。

通过将 8 位定时器模式控制寄存器 5n（TMC5n）的第 0 位（TOE5n）设置为 1，可以按照指定的间隔时间反转 TO5n 引脚输出状态，该间隔时间由 8 位定时器比较寄存器 5n（CR5n）的预置值决定。这样可以使能输出任选频率的方波（占空比＝50％）。

设置

<1> 设置各个寄存器

- 将端口输出锁存器（P44 或 P43）[注]和端口模式寄存器（PM44 或 PM43）[注]清除为 0。
- TCL5n：　选择计数时钟
- CR5n：　比较值
- TMC5n：　停止计数操作，选择 TM5n 与 CR5n 匹配时发生清零&启动的模式

LVS5n	LVR5n	定时器输出 F/F 状态设置
1	0	定时器输出 F/F 清除（0）（TO5n 输出的默认值：低电平）
0	1	定时器输出 F/F 置位（1）（TO5n 输出的默认值：高电平）

定时器输出使能
（TMC5n = 00001011B 或 00000111B）

<2> 在设置 TCE5n = 1 之后，计数操作开始。

<3> TM5n 与 CR5n 的匹配会反转定时器输出 F/F。产生 INTTM5n 后，TM5n 被清除为 00H。

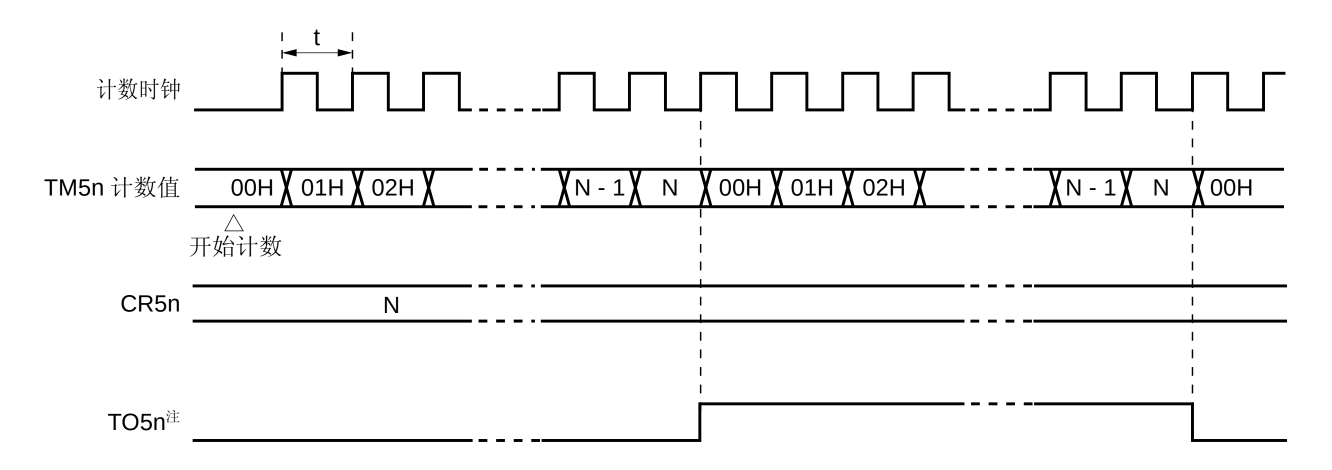
<4> 在经过上述设置后，按照相同的间隔时间反转定时器输出 F/F，并从 TO5n 输出方波。
频率如下。
　频率 = 1/2t（N + 1）
　（N：　00H 至 FFH）

注　8 位定时器/事件计数器 50：P44，PM44
　　8 位定时器/事件计数器 51：P43，PM43

注意事项　操作期间，不要向 CR5n 写入其它值。

备注　**1.** 如需了解如何使能 INTTM5n 中断，参见**第二十章 中断功能**。
　　　2. n = 0，1

图 7-17. 方波输出的操作时序



注　TO5n 输出的初始值由 8 位定时器模式控制寄存器 5n（TMC5n）的第 2 位和第 3 位（LVR5n，LVS5n）设置。

7.4.4 PWM输出操作

当 8 位定时器模式控制寄存器 5n（TMC5n）的第 6 位（TMC5n6）被设置为 1 时，8 位定时器/事件计数器 5n 作为 PWM 输出操作。

占空比由 8 位定时器比较寄存器 5n（CR5n）的设置值决定，并从 TO5n 输出。

将 PWM 脉冲的有效电平宽度设置到 CR5n；有效电平可以由 TMC5n 的第 1 位（TMC5n1）选择。

由定时器时钟选择寄存器 5n（TCL5n）的第 0 位至第 2 位（TCL5n0 至 TCL5n2）来选择计数时钟。

可以由 TMC5n 的第 0 位（TOE5n）决定 PWM 输出的使能/禁止。

注意事项　在 PWM 模式下，重写 CR5n 的间隔至少要大于计数时钟（由 TCL5n 选择时钟）的 3 个周期。

备注　n = 0，1

（1）　PWM 输出的基本操作

设置

<1>　设置各个寄存器
- 将端口输出锁存器（P44 或 P43）注 和端口模式寄存器（PM44 或 PM43）注 清除为 0。
- TCL5n：　选择计数时钟
- CR5n：　比较值
- TMC5n：　停止计数操作，选择 PWM 模式
  定时器输出 F/F 未改变。

TMC5n1	有效电平选择
0	高电平有效
1	低电平有效

定时器输出使能
（TMC5n = 01000001B 或 01000011B）

<2>　当 TCE5n = 1 时，计数操作开始。
将 TCE5n 清除为 0，可以停止计数操作。

注　8 位定时器/事件计数器 50：　P44，PM44
8 位定时器/事件计数器 51：　P43，PM43

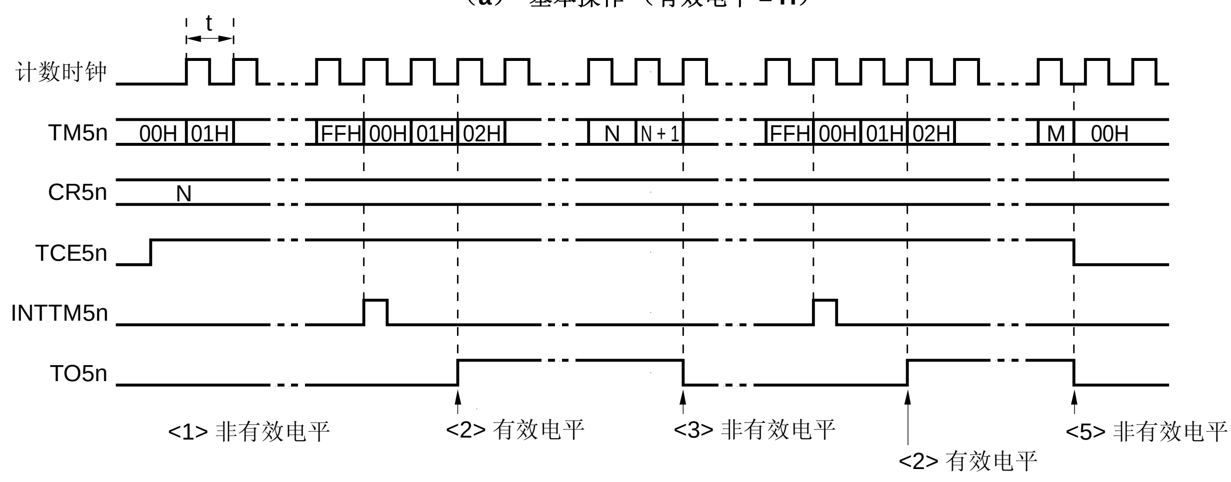
PWM 输出操作

<1>　PWM 输出（从 TO5n 输出）会持续输出非有效电平，直至发生溢出。
<2>　当发生溢出时，输出有效电平。有效电平会持续输出，直至 CR5n 与 8 位定时器计数器 5n（TM5n）的计数值匹配时。
<3>　在 CR5n 与计数值匹配之后，将会持续输出非有效电平，直至再次发生溢出。
<4>　重复 <2>和 <3>的操作，直至计数操作停止。
<5>　通过 TCE5n = 0 停止计数操作时，PWM 的输出变为非有效电平。
需要了解时序的详细信息，参见图 **7-18** 和 **7-19**。
周期、有效电平宽度和占空比如下。
- 周期 = 2^8t
- 有效电平宽度 = Nt
- 占空比 = $N/2^8$
  （N = 00H 至 FFH）

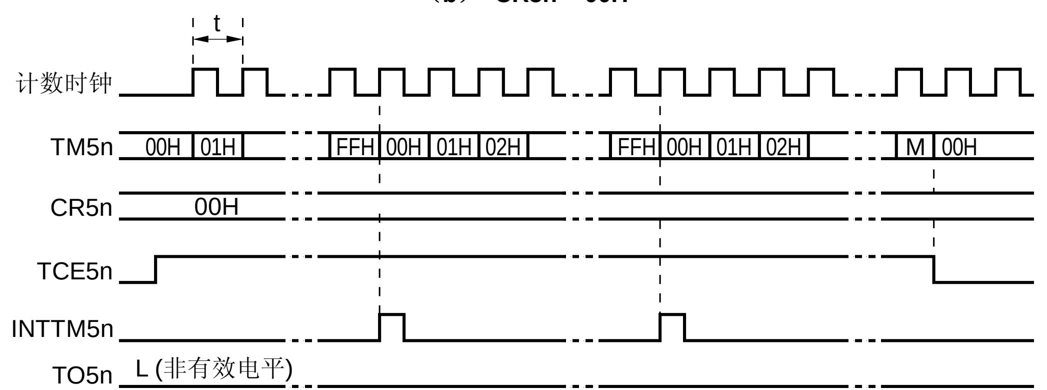
备注　n = 0，1

图 7-18. PWM 输出操作的时序

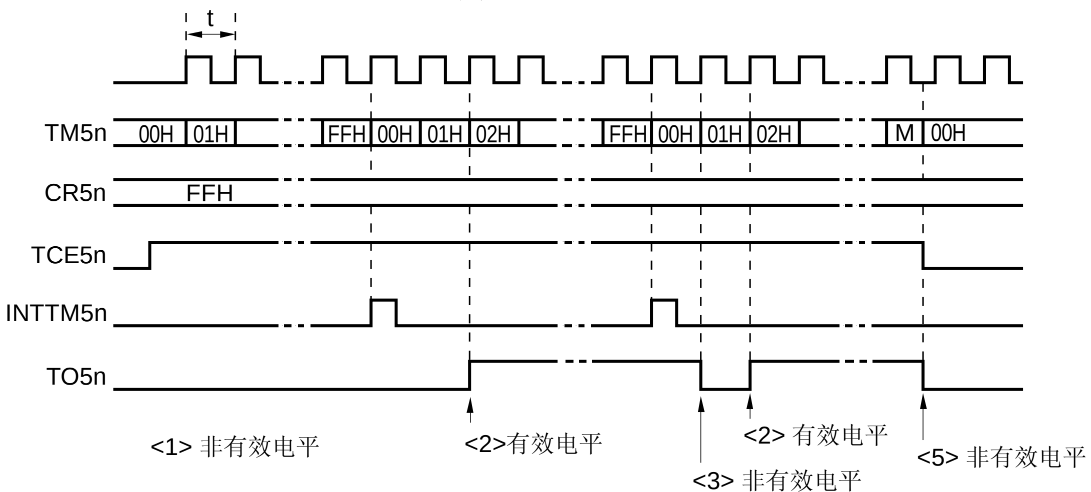
（a） 基本操作 （有效电平 = H）



（b） CR5n = 00H



（c） CR5n = FFH



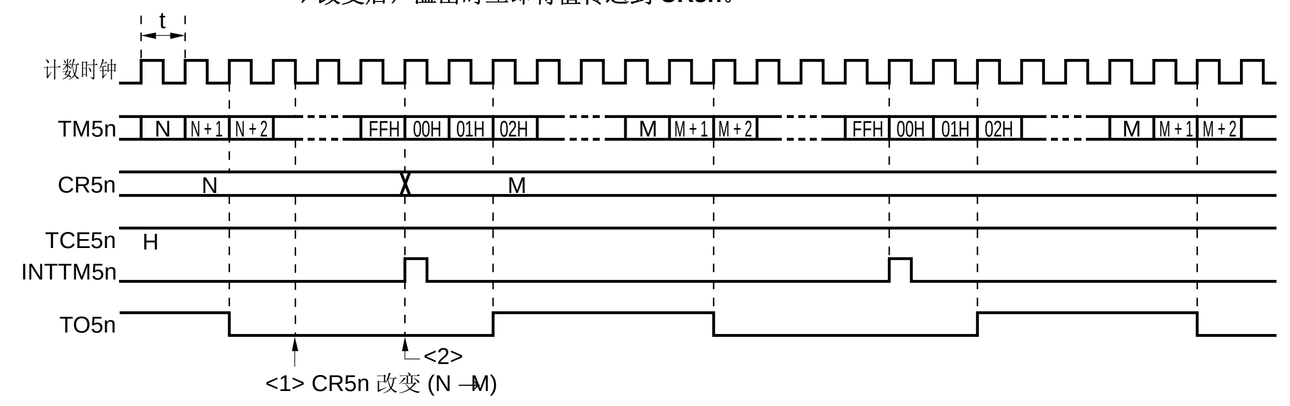
备注 **1.** 图 7-18（a）中的<1>至<3> 和 <5>步骤对应于 **7.4.4 （1） PWM 输出基本操作**中 PWM 输出操作 的<1>至<3> 和 <5>步骤。

2. n = 0，1

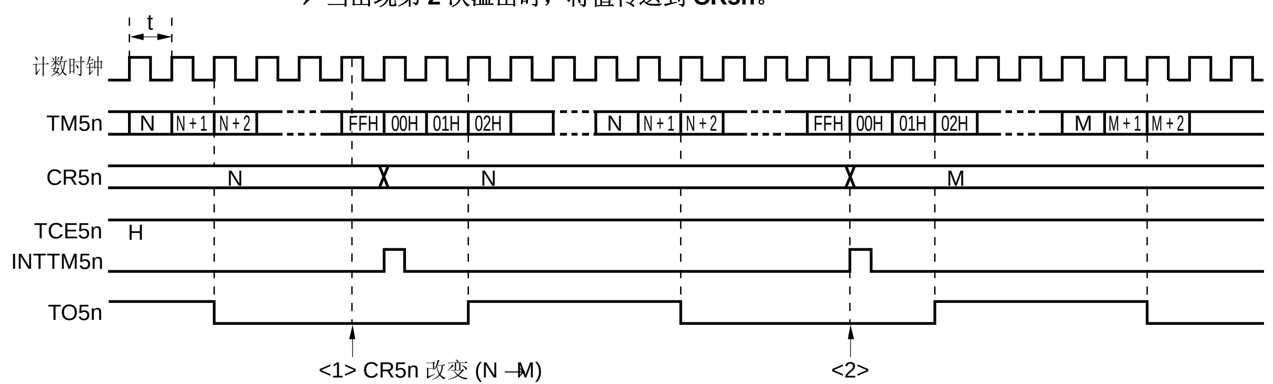
（**2**） 当 **CR5n** 改变时的操作

图 **7-19.** 改变 **CR5n** 时的操作时序

（**a**） 在计数值 **FFH** 的时钟上升沿之前，**CR5n** 的值由 **N** 改变为 **M**
→ 改变后，溢出时立即将值传送到 **CR5n**。



（**b**） 在计数值 **FFH** 时的时钟上升沿之后，**CR5n** 的值由 **N** 变为 **M**
→ 当出现第 **2** 次溢出时，将值传送到 **CR5n**。



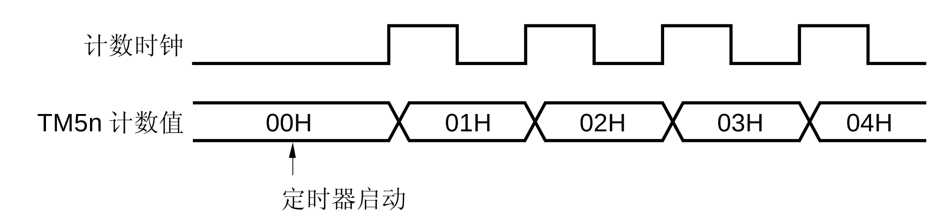
注意事项 在图 **7-19** 中的**<1>**与**<2>**之间读取 **CR5n** 时，读取的值与实际值不同（读取的值：**M**，**CR5n** 实际值：**N**）。

7.5 使用 8 位定时器/事件计数器 50、51 和 52 的注意事项

（**1**） 定时器启动误差

定时器启动后，在产生匹配信号时可能产生最大一个时钟周期的误差。这是因为 8 位定时器计数器 50、51 和 52（TM50，TM51 和 TM52）计数操作的启动和计数脉冲不同步引起的。

图 7-20. 8 位定时器计数器 5n 启动时序



备注　　n = 0 至 2

（2） 外部 24 位事件计数器操作期间，16 位定时器/事件计数器 00 向上计数的注意事项

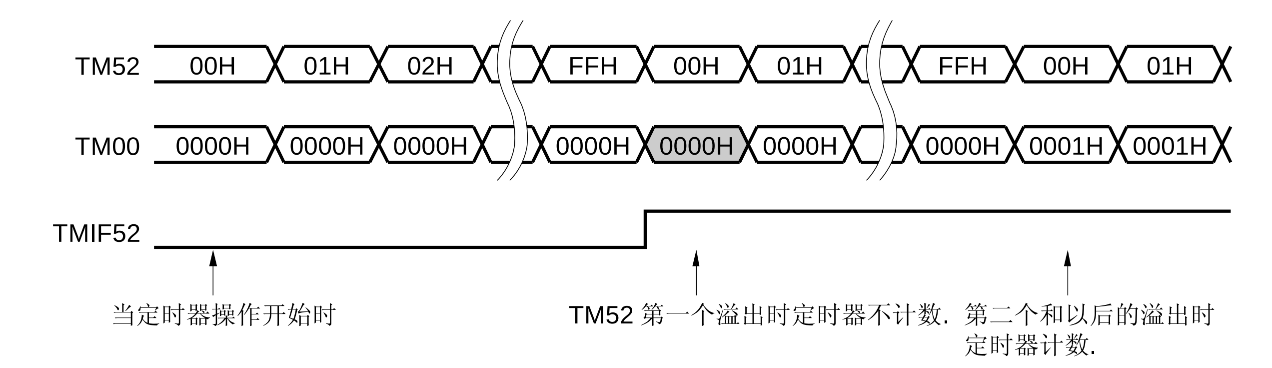
16 位定时器/事件计数器 00 具有内部同步电路，当开始操作时可以消除噪声，操作开始后的并不对第一个时钟立即计数。

通过设置 16 位定时器/事件计数器 00 和 8 位定时器/事件计数器 52 作为高位和低位定时器，并将它们级联，该计数器作为 24 位计数器使用，作为低位定时器的 8 位定时器/事件计数器 52 的中断请求标志必须按照下面的描述进行检查，以便准确读出 24 位计数值。

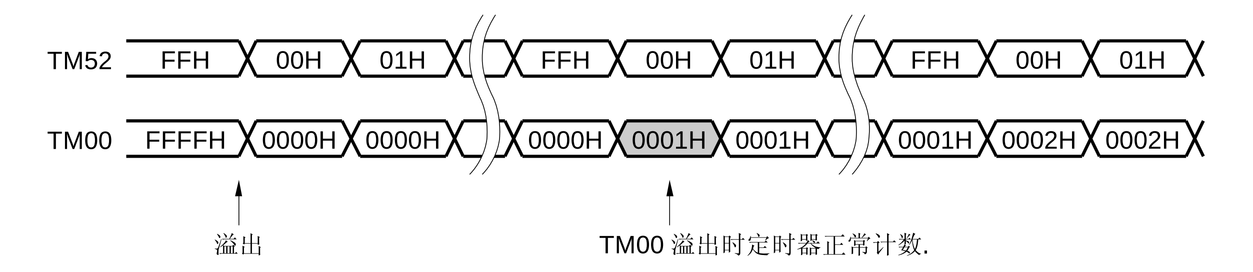
- 如果读取 TM52 和 TM00 时，TMIF52 = 1：
  TM00 实际的计数值是“TM00 的读取值 + 1”。
- 如果读取 TM52 和 TM00 时，TMIF52 = 0：
  读取的值是正确的值。

仅当操作开始后，16 位定时器/事件计数器 00 会出现这种现象。当 16 位定时器/事件计数器 00 溢出且从 0000H 重启计数时，不发生计数延迟，因为已经实现同步。

<当操作启动时>



<高位定时器的溢出>



<R>　（3）　**8 位定时器计数器 5n（TM5n）的读取**

无需停止实际的计数器就可以读取 TM5n，因为读取已经被捕获到缓冲器中的计数值时，该值是固定的。但是，在计数起向上计数之前的瞬间读取时，该缓冲器可能还没有被更新，因为该缓冲器是在计数器向上计数的时序时被更新。

图 7-21. 16 位定时器计数器 5n（TM5n）读取时序

备注　n = 0 至 2

8.1 8 位定时器 H0、H1 和H2 的功能

8 位定时器/事件计数器 H0、H1 和 H2 具有以下功能

- 间隔定时器
- 方波输出[注 1]
- PWM 输出[注 2]
- 载波发生器（仅用于 8 位定时器 H1）[注 3]

注 **1.** 仅限 TMH0 和 TMH1 。
2. 但是，TOHn 只有 TOH0 和 TOH1。
3. 仅限 TMH1 。TM51 和 TMH1 可组合用作载波发生器模式。

8.2 8 位定时器 H0、H1 和H2 的配置

8 位定时器 H0、H1 和 H2 包括以下硬件。

表 8-1. 8 位定时器 H0、H1 和 H2 的配置

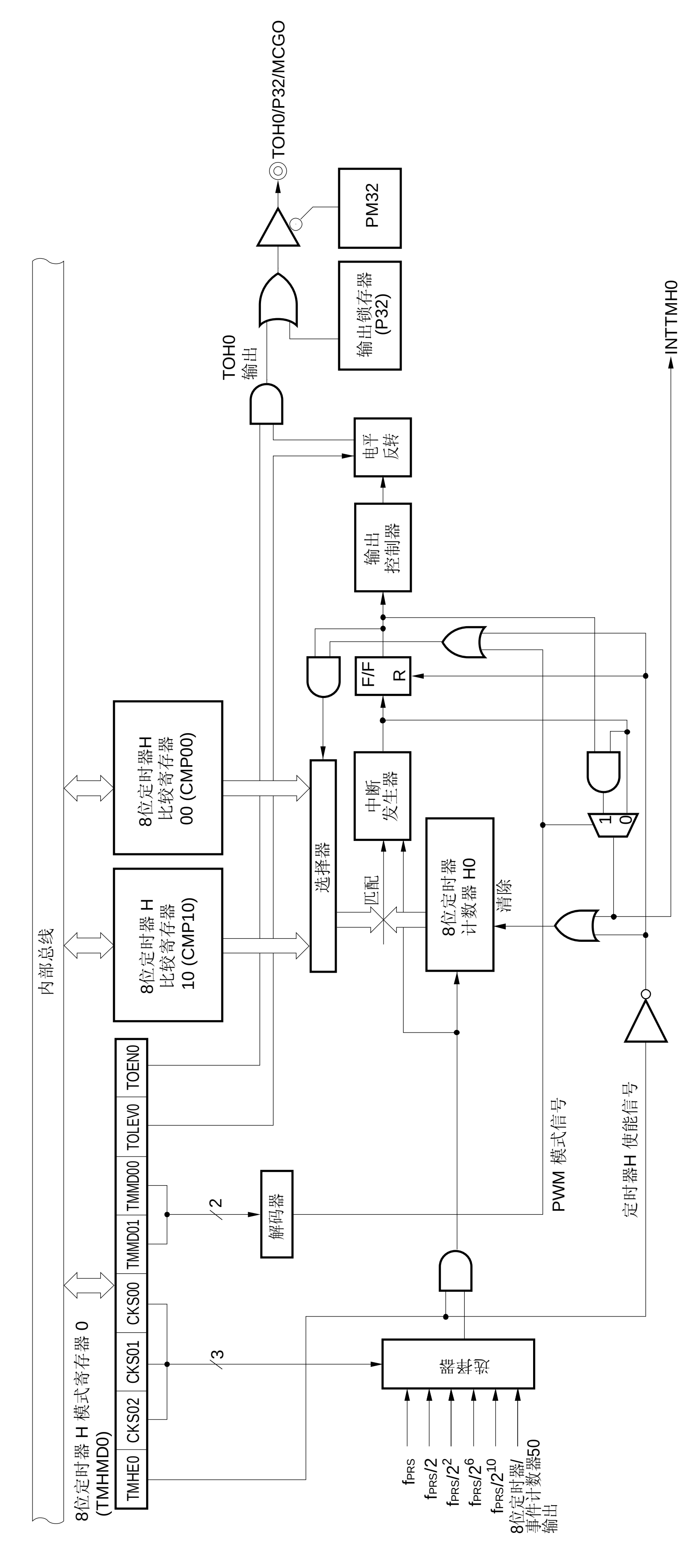
项目	配置
定时器寄存器	8 位定时器计数 Hn
寄存器	8 位定时器 H 比较寄存器 0n （CMP0n） 8 位定时器 H 比较寄存器 1n （CMP1n））
定时器输出	TOHn[注 1]，输出控制器
控制寄存器	8 位定时器 H 模式寄存器 n （TMHMDn） 8 位定时器 H 载波控制寄存器 1 （TMCYC1）[注 2] 端口模式寄存器 3（PM3） 端口寄存器 3（P3）

注 **1.** TMH2 没有输出引脚（TOH2）。仅可用作内部中断（INTTMH2）或 TI52 引脚的外部事件输入使能信号。
2. 仅限 8 位定时器 H1

备注 n = 0 至 2。TOHn 只有 TOH0 和 TOH1 。

图 8-1 至 8-3 为定时器框图。

图 8-1. 8 位定时器 H0 的框图



<R>

图 8-2. 8 位定时器 H1 的框图

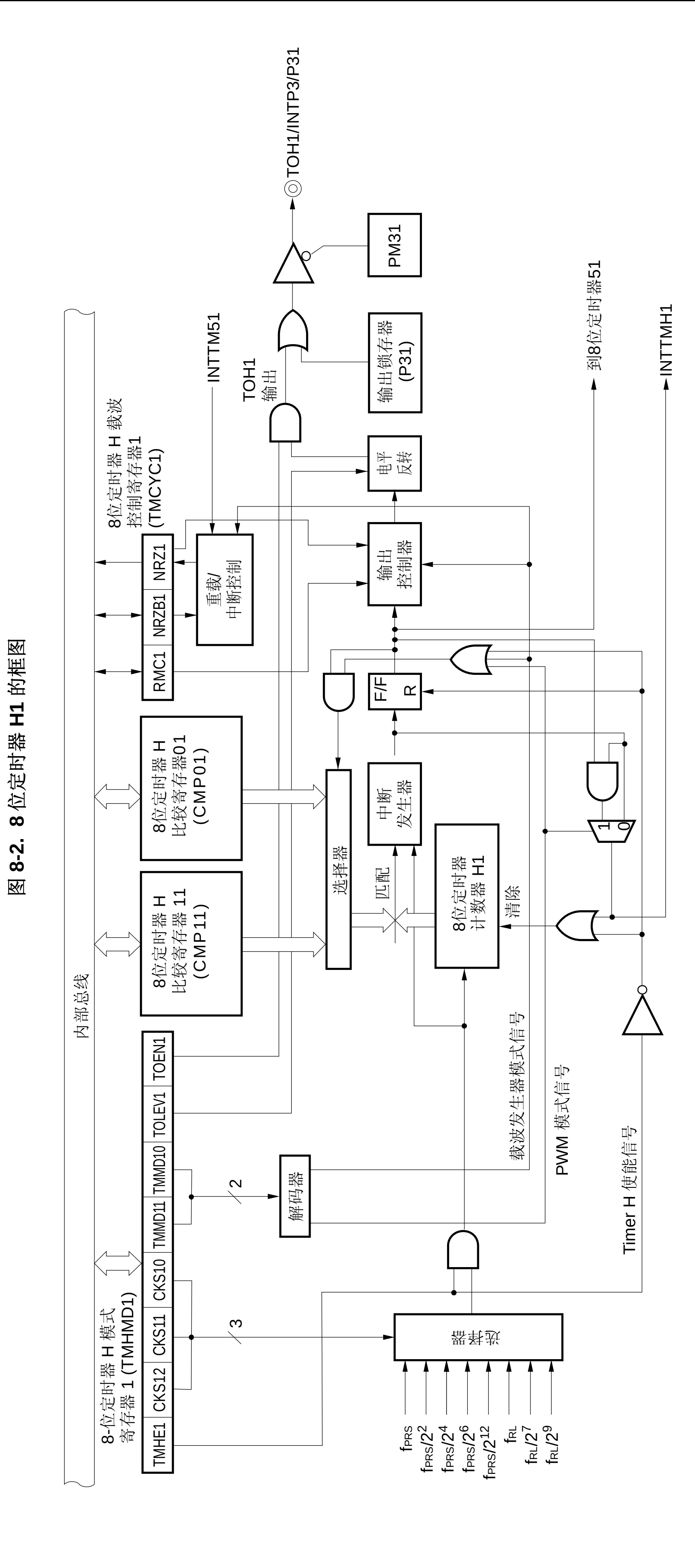
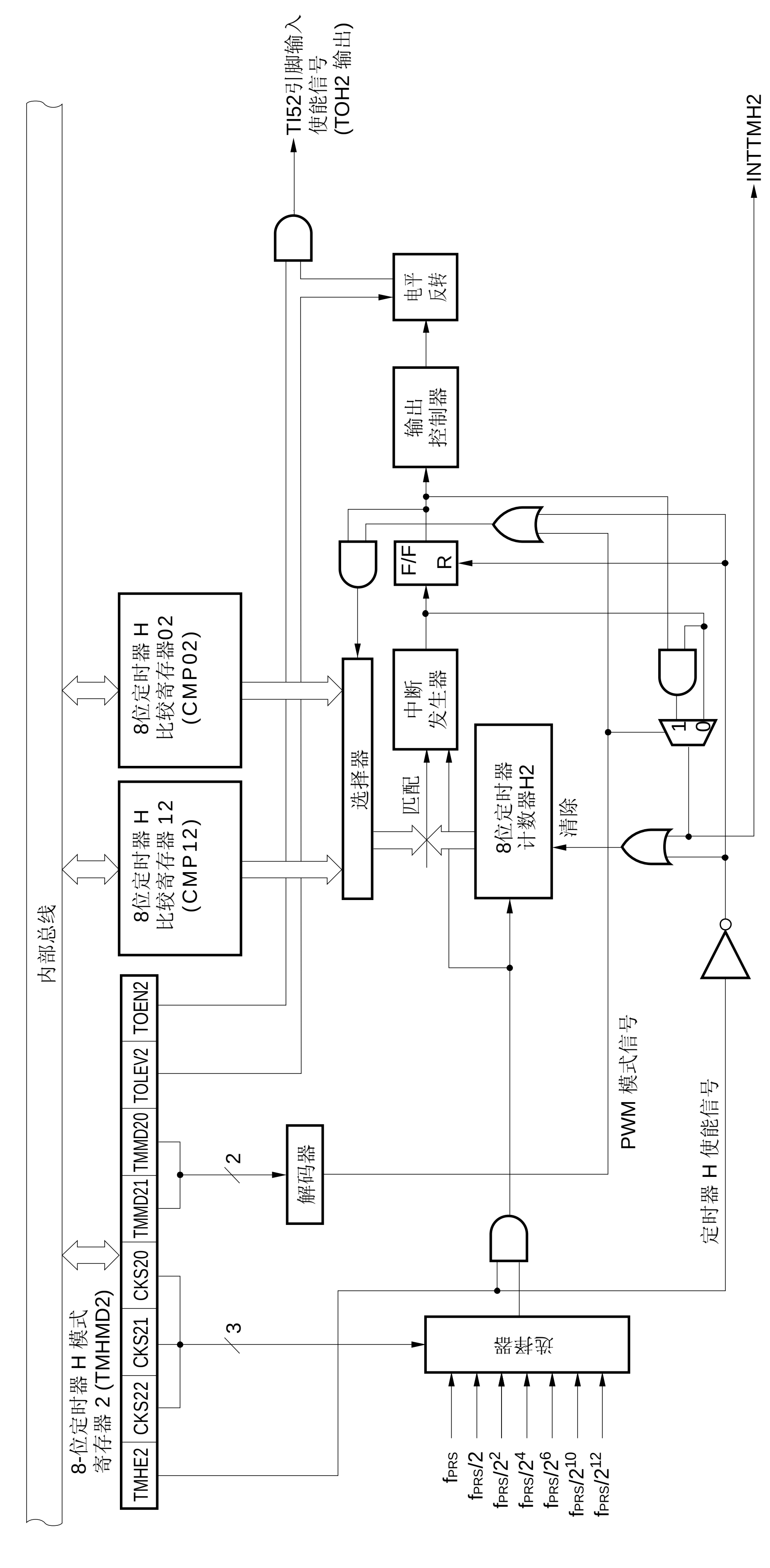


图 8-3. 8 位定时器 H2 的框图



（1） 8 位定时器 H 比较寄存器 0n（CMP0n）

该寄存器可以通过 8 位存储器操作指令进行读取或写入。该寄存器可用于所有的定时器操作模式。

该寄存器总是将 CMP0n 的值和 8 位定时器计数器 Hn 的计数值进行比较，当两个值匹配时，产生一个中断请求信号（INTTMHn）并反转 TOHn 的输出电平。

当定时器停止时（TMHEn = 0），重写 CMP0n 的值。

复位信号的产生会将该寄存器设置为 00H。

图 8-4. 8 位定时器 H 比较寄存器 0n（CMP0n）的格式

地址： FF18H (CMP00)， FF1AH (CMP01)， FF44H (CMP02) 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
CMP0n (n = 0 至 2)								

注意事项 在定时器计数操作期间，不能重写 CMP0n。在定时器计数操作期间，可以刷新 CMP0n（写入相同的值）。

（2） 8 位定时器 H 比较寄存器 1n（CMP1n）

该寄存器可以通过 8 位存储器操作指令进行读取或写入。该寄存器可用于 PWM 输出模式和载波发生器模式。

在 PWM 输出模式下，该寄存器总是将 CMP1n 的设置值和 8 位定时器计数器 Hn 的计数值进行比较，当两个值匹配时反转 TOHn 的输出电平。不产生中断请求信号。

在载波发生器模式下，CMP1n 寄存器总是将 CMP1n 的设置值和 8 位定时器计数器 Hn 的计数值进行比较，如果两个值匹配，则产生一个中断请求信号（INTTMHn）。同时，计数值被清除。

在定时器计数期间，可以重写 CMP1n。

如果在定时器操作时重写 CMP1n 的值，则当定时器的计数值和 CMP1n 以前的值匹配时，锁存新的值并传送到 CMP1n，然后 CMP1n 的值被改变为新值。如果计数值和 CMP1n 值的匹配与写入 CMP1n 的操作发生冲突，那么 CMP1n 的值将不会改变。

复位信号的产生会将该寄存器清除为 00H。

图 8-5. 8 位定时器 H 比较寄存器 1n 的格式（CMP1n）

地址： FF19H (CMP10)， FF1BH (CMP11)， FF45H (CMP12) 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
CMP1n (n = 0 至 2)								

注意事项 在 PWM 输出模式和载波发生器模式下，当停止定时器计数操作（TMHEn = 0）之后，启动定时器计数操作（TMHEn = 1）时，必须设置 CMP1n（即使将相同的值设置到 CMP1n，也必须再次设置）。

备注 n = 0 至 2。但是，TOHn 只有 TOH0 和 TOH1 。

8.3 控制 8 位定时器H0、H1 和H2 的寄存器

以下四个寄存器可以用来控制 8 位定时器 H0、H1 和 H2。

- 8 位定时器 H 模式寄存器 n （TMHMDn）
- 8 位定时器 H 载波控制寄存器 1 （TMCYC1）[注]
- 端口模式寄存器 3（PM3）
- 端口寄存器 3（P3）

注　　仅限 8 位定时器 H1。

（1） 8 位定时器 H 模式寄存器 n （TMHMDn）

该寄存器用于控制定时器 H 的模式。

可以由 1 位或 8 位存储器操作指令来设置该寄存器。

复位信号的产生会将该寄存器清除为 00H。

备注　　n = 0 至 2

图 8-6. 8 位定时器 H 模式寄存器 0（TMHMD0）的格式

地址: FF69H 复位后: 00H R/W

	<7>	6	5	4	3	2	<1>	<0>
TMHMD0	TMHE0	CKS02	CKS01	CKS00	TMMD01	TMMD00	TOLEV0	TOEN0

TMHE0	是否允许定时器操作
0	停止定时器计数操作(计数器清零)
1	允许定时器计数(由输入时钟启动计数操作)

CKS02	CKS01	CKS00	计数时钟选择[注 1]			
				f_{PRS} = 2 MHz	f_{PRS} = 5 MHz	f_{PRS} = 10 MHz
0	0	0	f_{PRS}[注2]	2 MHz	5 MHz	10 MHz
0	0	1	$f_{PRS}/2$	1 MHz	2.5 MHz	5 MHz
0	1	0	$f_{PRS}/2^2$	500 kHz	1.25 MHz	2.5 MHz
0	1	1	$f_{PRS}/2^6$	31.25 kHz	78.13 kHz	156.25 kHz
1	0	0	$f_{PRS}/2^{10}$	1.95 kHz	4.88 kHz	9.77 kHz
1	0	1	TM50 输出[注 3]			
其它情况			禁止设置			

<R>

TMMD01	TMMD00	定时器操作模式
0	0	间隔定时器模式
1	0	PWM 模式
其它情况		禁止设置

TOLEV0	定时器输出电平控制 (默认模式)
0	低电平
1	高电平

TOEN0	定时器输出控制
0	禁止输出
1	允许输出

注 **1.** 如果外设硬件时钟（f_{PRS}）使用高速系统时钟（f_{XH}）（XSEL = 1），f_{PRS} 操作频率会根据供电电压不同而变动。

- V_{DD} = 2.7 至 5.5 V： $f_{PRS} \leq 10$ MHz
- V_{DD} = 1.8 至 2.7 V： $f_{PRS} \leq 5$ MHz

2. 如果外设硬件时钟（f_{PRS}） 使用内部高速振荡时钟（f_{XH}）（XSEL = 0），当 1.8 V ≤ V_{DD} < 2.7 V 时，禁止设置 CKS02 = CKS01 = CKS00 = 0（计数时钟：f_{PRS}）。

注　**3.** 选择 TM50 输出作为计数时钟时，应注意以下几点。
- 在 TM50 和 CR50 的值匹配时计数时钟被清除且启动的模式（TMC506 = 0），
  先启动 8 位定时器/事件计数器 50，然后使能定时器 F/F 反转操作（TMC501 = 1）
- PWM 模式（TMC506 = 1）
  先启动 8 位定时器/事件计数器 50 的操作，然后设置计数时钟，使占空比= 50%

在任何模式中都无需使能（TOE50 = 1）TO50 输出。

注意事项　1. **当 TMHE0=1 时，禁止设置 TMHMD0 的其他位。但是，可以刷新 TMHMD0（写入相同值）。**

2. 在 PWM 输出模式下，当停止定时器计数操作（TMHE0 = 0）之后，启动定时器计数操作（TMHE0 = 1）时，必须设置 8 位定时器 H 比较寄存器 10（CMP10）（即使将相同的值设置到 CMP10，也必须再次设置）。

3. 除 TOH0 的输出以外，TOH0/P32/MCGO 引脚的实际输出由 PM32 和 P32 决定。

备注　**1.** f_{PRS}：外设硬件时钟频率

2. TMC506：8 位定时器模式控制寄存器 50（TMC50）的第 6 位
TMC501：TMC50 的第 1 位

图 8-7. 8 位定时器 H 模式寄存器 1（TMHMD1）的格式

地址: FF6CH 复位后: 00H R/W

	<7>	6	5	4	3	2	<1>	<0>
TMHMD1	TMHE1	CKS12	CKS11	CKS10	TMMD11	TMMD10	TOLEV1	TOEN1

TMHE1	是否允许定时器操作
0	停止定时器计数操作 (计数器清零)
1	允许定时器计数 (由输入时钟启动计数操作)

CKS12	CKS11	CKS10	计数时钟选择[注 1]			
				f_{PRS} = 2 MHz	f_{PRS} = 5 MHz	f_{PRS} = 10 MHz
0	0	0	f_{PRS}[注 2]	2 MHz	5 MHz	10 MHz
0	0	1	$f_{PRS}/2^2$	500 kHz	1.25 MHz	2.5 MHz
0	1	0	$f_{PRS}/2^4$	125 kHz	312.5 kHz	625 kHz
0	1	1	$f_{PRS}/2^6$	31.25 kHz	78.13 kHz	156.25 kHz
1	0	0	$f_{PRS}/2^{12}$	0.49 kHz	1.22 kHz	2.44 kHz
1	0	1	$f_{RL}/2^7$	1.88 kHz (TYP.)		
1	1	0	$f_{RL}/2^9$	0.47 kHz (TYP.)		
1	1	1	f_{RL}	240 kHz (TYP.)		

TMMD11	TMMD10	定时器操作模式
0	0	间隔定时器模式
0	1	载波发生器模式
1	0	PWM 输出模式
1	1	禁止设置

TOLEV1	定时器输出电平控制模式(默认模式)
0	低电平
1	高电平

TOEN1	定时器输出控制
0	禁止输出
1	允许输出

注 **1.** 如果外设硬件时钟（f_{PRS}）使用高速系统时钟（f_{XH}）（XSEL ＝ 1），f_{PRS} 操作频率会根据供电电压不同而变动。

- V_{DD} = 2.7 至 5.5 V： $f_{PRS} \leq 10$ MHz
- V_{DD} = 1.8 至 2.7 V： $f_{PRS} \leq 5$ MHz

2. 如果外设硬件时钟（f_{PRS}） 使用内部高速振荡时钟（f_{XH}）（XSEL = 0），当 1.8 V ≤ V_{DD} < 2.7 V 时，禁止设置 CKS12 = CKS11 = CKS10 = 0（计数时钟：f_{PRS}）。

注意事项 1. 当 **TMHE1 = 1** 时，禁止设置 **TMHMD1** 的其他位。但是，可以刷新 **TMHMD1**（写入相同值）。

2. 在 **PWM** 输出模式下，当停止定时器计数操作（**TMHE1 = 0**）之后，启动定时器计数操作（**TMHE1 = 1**）时，必须设置 **8** 位定时器 **H** 比较寄存器 **11**（**CMP11**）（即使将相同的值设置到 **CMP11**，也必须再次设置）。

3. 使用载波发生器模式时，**TMH1** 的计数时钟频率要设置为 **TM51** 的计数时钟频率的 **6** 倍以上。

4. 除 **TOH1** 的输出以外，**TOH1/P31/INTP3** 引脚的实际输出由 **PM31** 和 **P31** 决定。

备注 1. f_{PRS}： 外设硬件时钟频率

2. f_{RL}： 内部低速振荡时钟频率

图 8-8. 8 位定时器 H 模式寄存器 2（TMHMD2）的格式

地址: FF42H　复位后: 00H　R/W

	<7>	6	5	4	3	2	<1>	<0>
TMHMD2	TMHE2	CKS22	CKS21	CKS20	TMMD21	TMMD20	TOLEV2	TOEN2

TMHE2	是否允许定时器操作
0	停止定时器计数操作 (计数器清零)
1	允许定时器计数操作(由输入时钟启动计数操作)

CKS22	CKS21	CKS20	计数时钟选择注 1			
				f_{PRS} = 2 MHz	f_{PRS} = 5 MHz	f_{PRS} = 10 MHz
0	0	0	f_{PRS}注 2	2 MHz	5 MHz	10 MHz
0	0	1	$f_{PRS}/2$	1 MHz	2.5 MHz	5 MHz
0	1	0	$f_{PRS}/2^2$	500 kHz	1.25 MHz	2.5 MHz
0	1	1	$f_{PRS}/2^4$	125 kHz	312.5 kHz	625 kHz
1	0	0	$f_{PRS}/2^6$	31.25 kHz	78.13 kHz	156.25 kHz
1	0	1	$f_{PRS}/2^{10}$	1.95 kHz	4.88 kHz	9.77 kHz
1	1	0	$f_{PRS}/2^{12}$	0.49 kHz	1.22 kHz	2.44 kHz
其它情况			禁止设置			

TMMD21	TMMD20	定时器操作模式
0	0	间隔定时器模式
1	0	引脚输入允许宽度判断模式 (PWM 模式)
其它情况		禁止设置

TOLEV2	定时器输出电平控制 (默认模式)
0	低电平
1	高电平

TOEN2	定时器输出控制
0	禁止输出
1	允许输出注 3

注　**1.** 如果外设硬件时钟（f_{PRS}）使用高速系统时钟（f_{XH}）（XSEL = 1），f_{PRS} 操作频率会根据供电电压不同而变动。

- V_{DD} = 2.7 至 5.5 V：　$f_{PRS} \leq 10$ MHz
- V_{DD} = 1.8 至 2.7 V：　$f_{PRS} \leq 5$ MHz

2. 如果外设硬件时钟（f_{PRS}） 使用内部高速振荡时钟（f_{XH}）（XSEL = 0），当 1.8 V ≤ V_{DD} < 2.7 V 时，禁止设置 CKS22 = CKS21 = CKS20 = 0（计数时钟：f_{PRS}）。

3. TMH2 的定时器输出只能用作 TM52 的外部事件输入使能信号。不为外部输出提供引脚。

注意事项　当 TMHE2 = 1 时，禁止设置 TMHMD2 其他位。

备注　f_{PRS}：外设硬件时钟频率

（2） 8 位定时器 H 载波控制寄存器 1 （TMCYC1）

该寄存器用于控制 8 位定时器 H1 的遥控输出和载波脉冲输出状态。

可以由 1 位或 8 位存储器操作指令来设置该寄存器。

复位信号的产生会将该寄存器清除为 00H。

图 8-9. 8 位定时器 H 载波控制寄存器 1（TMCYC1）的格式

地址： FF6DH 复位后： 00H R/W[注]

符号	7	6	5	4	3	2	1	<0>
TMCYC1	0	0	0	0	0	RMC1	NRZB1	NRZ1

RMC1	NRZB1	遥控输出
0	0	低电平输出
0	1	在 INTTM51 信号输入上升沿高电平输出
1	0	低电平输出
1	1	在 INTTM51 信号输入上升沿载波脉冲输出

NRZ1	载波脉冲输出状态标志
0	禁止载波输出状态 (低电平状态)
1	允许载波输出状态 (RMC1 = 1： 载波脉冲输出 RMC1 = 0： 高电平状态)

注 第 0 位是只读的。

注意事项 当 TMHE1 = 1 时，不能重写 RMC1。但是，可以刷新 TMCYC1（写入相同的值）。

（3） 端口模式寄存器 3（PM3）

该寄存器按位设置端口 3 的输入输出模式。

当 P32/TOH0/MCGO 和 P31/TOH1/INTP3 引脚用于定时器输出时，清除 PM32 和 PM31 为 0，将 P32 和 P31 的输出锁存清除为 0。

可以由 1 位或 8 位存储器操作指令来设置 PM3 。

复位信号的产生会将寄存器设置为 FFH。

图 8-10. 端口模式寄存器 3（PM3）的格式

地址： FF23H 复位后： FFH R/W

符号	7	6	5	4	3	2	1	0
PM3	1	1	1	PM34	PM33	PM32	PM31	1

PM3n	P3n 引脚 I/O 模式选择（n = 1 至 4）
0	输出模式（输出缓冲器打开）
1	输入模式（输出缓冲器关闭）

8.4 8 位定时器H0、H1 和H2 的操作

8.4.1 间隔定时器/方波输出操作

当 8 位定时器计数器 Hn 与比较寄存器 0n（CMP0n）匹配时，产生中断请求信号（INTTMHn）并将 8 位定时器计数器 Hn 清除为 00H。

在间隔定时器模式下，不使用比较寄存器 1n（CMP1n）。即使 CMP1n 寄存器被设置，也不会检测 8 位定时器计数器 Hn 与 CMP1n 寄存器的匹配，所以定时器的输出不受影响。

将定时器 H 模式寄存器 n（TMHMDn）的第 0 位（TOENn）置 1，可以从 TOHn 输出任意频率的方波（占空比 = 50%）。

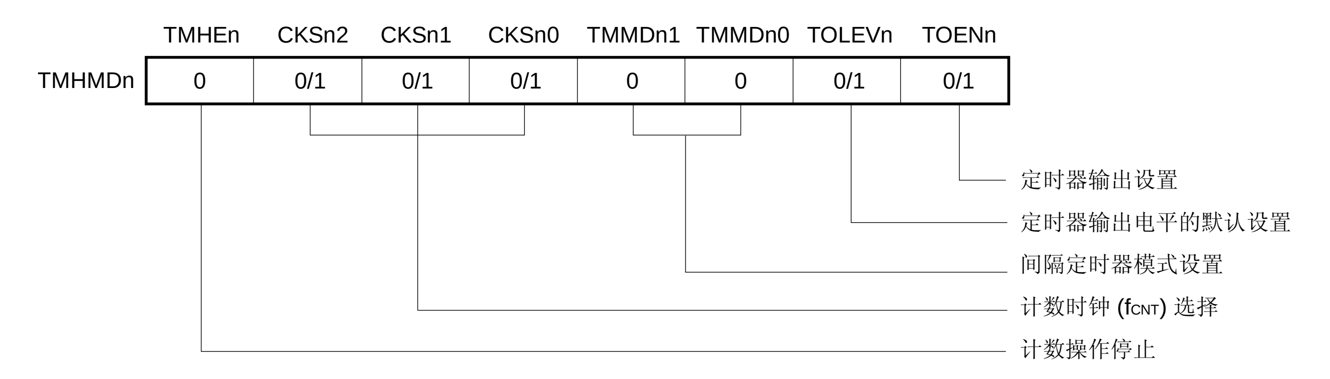
TMH2 的定时器输出只能用作 TM52 外部事件输入使能信号。请注意，不为外部输出提供引脚。

设置

<1> 设置各个寄存器。

图 8-11. 间隔定时器/方波输出操作期间的寄存器设置

（i） 设置定时器 H 模式寄存器 n （TMHMDn）



（ii） 设置 CMP0n 寄存器

如果 N 被设置作为比较值，则间隔时间如下：

- 间隔时间 = （N +1）/f_{CNT}

<2> 当 TMHEn = 1 时，计数操作开始。

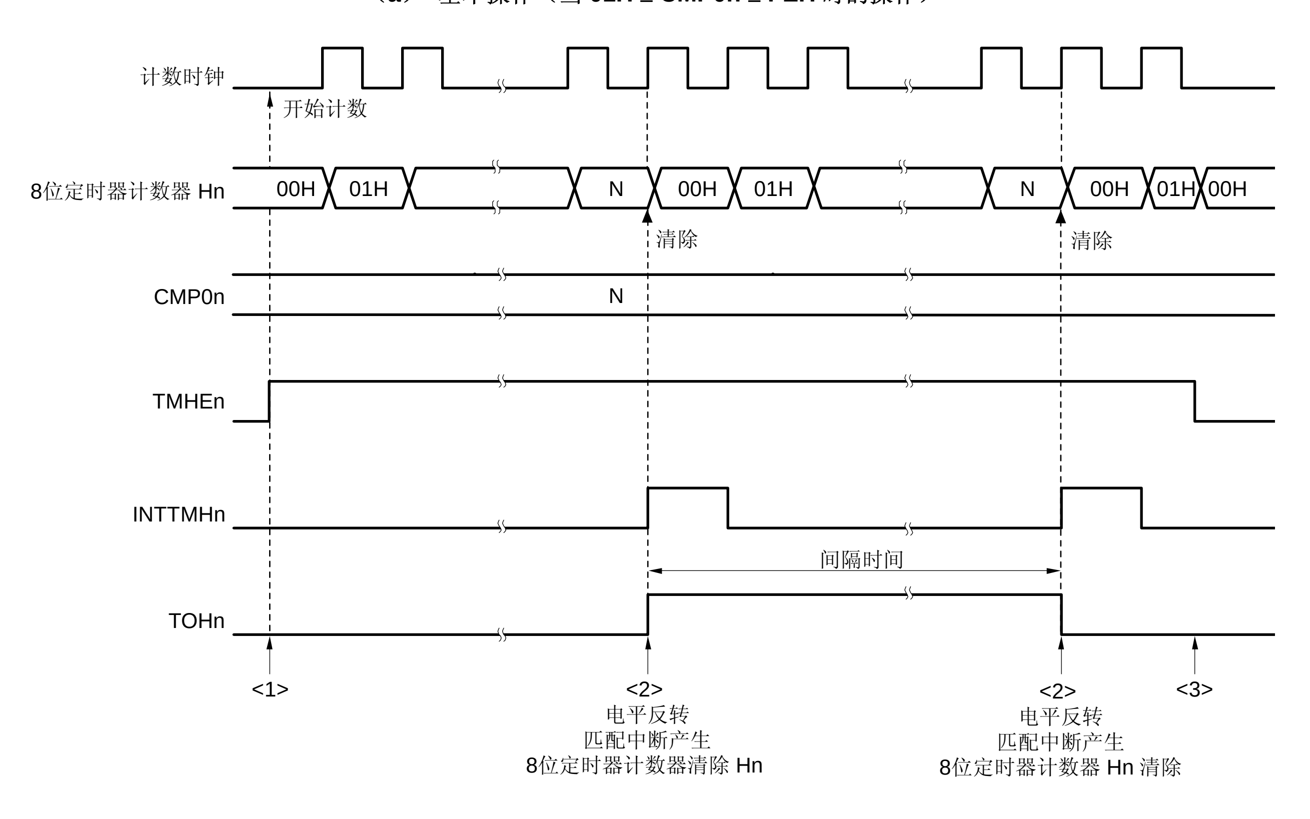
<3> 当 8 位定时器计数器 Hn 的值与 CMP0n 寄存器匹配时，产生 INTTMHn 信号，并将 8 位定时器计数器 Hn 清除为 00H。

<4> 随后，以相同的间隔时间重复产生 INTTMHn 信号。若要停止计数操作，将 TMHEn 清除为 0。

备注 **1.** 如需了解输出引脚的设置，参见 **8.3 （3） 端口模式寄存器 3（PM3）**。
2. 如需了解使能 INTTMHn 信号中断，参见 **第二十章 中断功能**。
3. n = 0 至 2。但是，TOHn 只有 TOH0 和 TOH1 。

图 8-12. 间隔定时器/方波输出的操作时序（1/2）

（a） 基本操作（当 01H ≤ CMP0n ≤ FEH 时的操作）



<1> 将 TMHEn 位设置为 1，使能计数操作。在操作使能之后，在不到 1 个时钟周期的时间内计数时钟开始计数。

<2> 当 8 位定时器计数器 Hn 的值与 CMP0n 寄存器的值匹配时，该定时器计数器的值被清除，且反转 TOHn 的输出电平。此外，在计数时钟的上升沿输出 INTTMHn 信号。

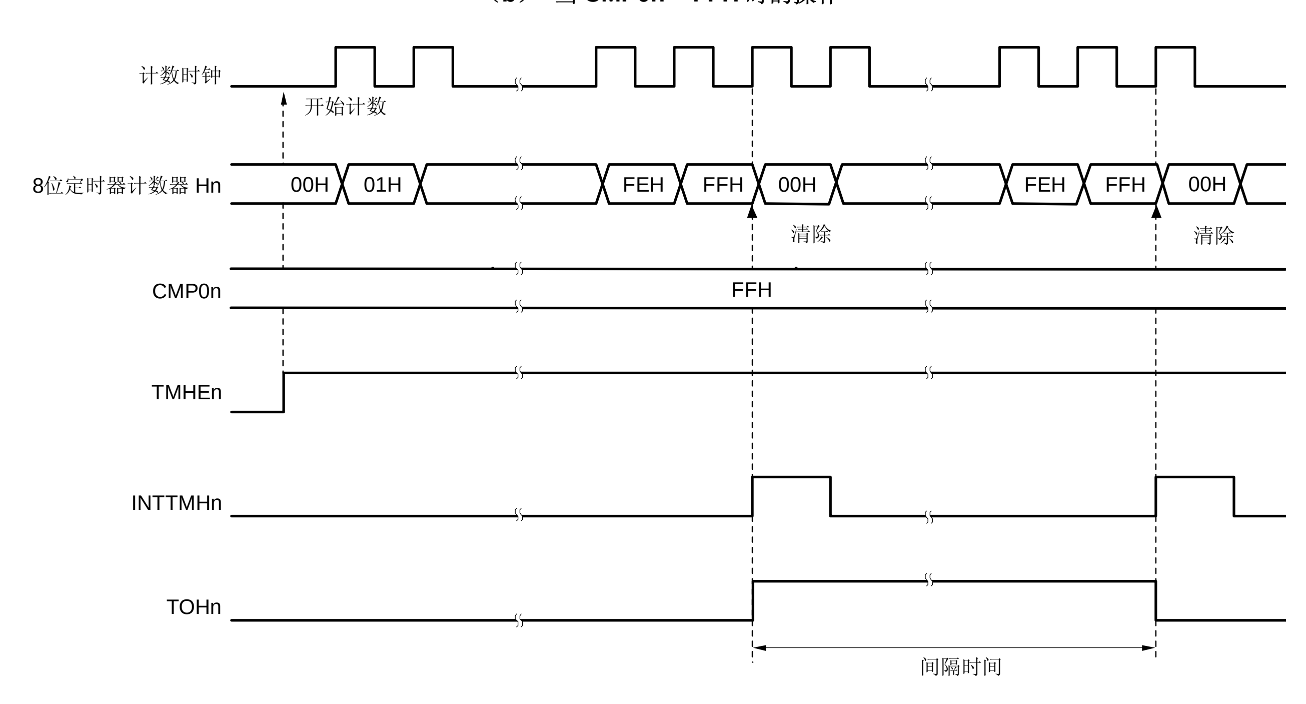
<3> 在定时器 Hn 操作时将 TMHEn 位清除为 0，INTTMHn 信号和 TOHn 输出被设置为默认电平。如果在将 TMHEn 位清除为 0 之前它们已经处于默认电平，则电平保持不变。

备注 **1.** n = 0 至 2。TOHn 只有 TOH0 和 TOH1。

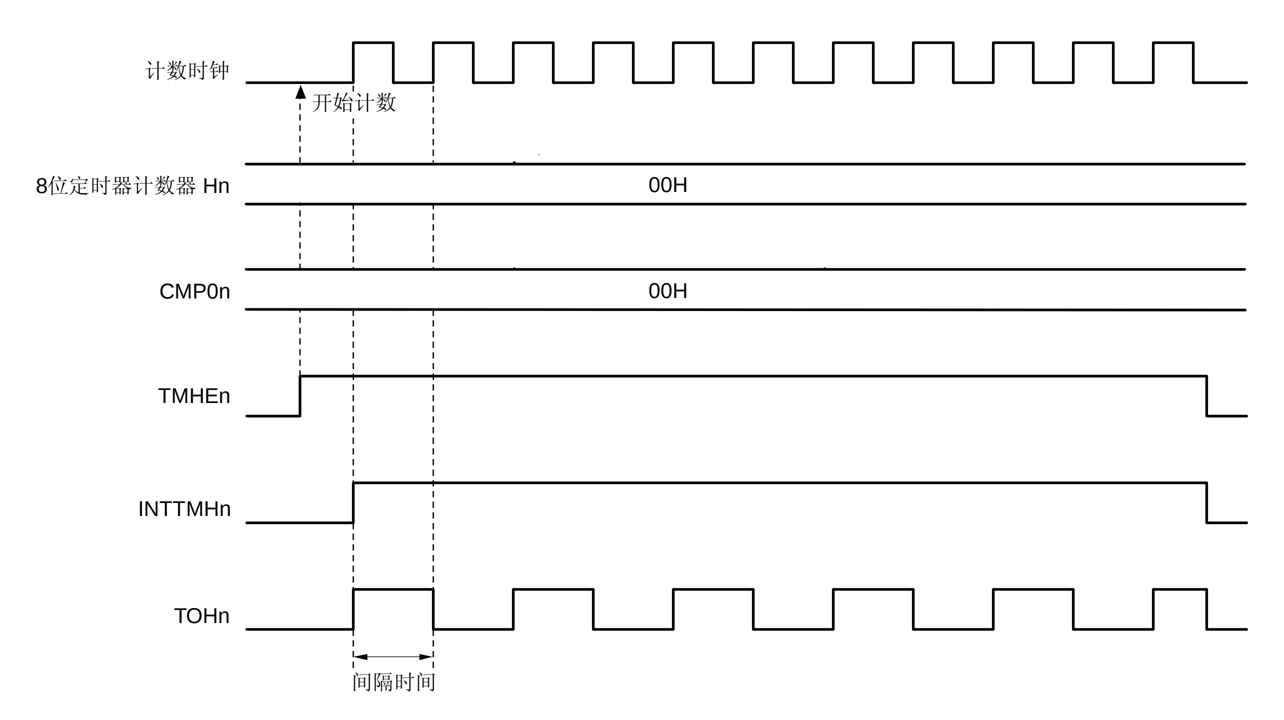
2. 01H ≤ N ≤ FEH

图 8-12. 间隔定时器/方波输出的操作时序（2/2）

（b） 当 CMP0n = FFH 时的操作



（c） 当 CMP0n = 00H 时的操作



备注 n = 0 至 2。TOHn 只有 TOH0 和 TOH1。

8.4.2 PWM输出操作

在 PWM 输出模式下，可以输出具有任意占空比和任意周期的脉冲。

8 位定时器比较寄存器 0n（CMP0n）用于控制定时器输出（TOHn）的周期。在定时器操作期间，禁止重写 CMP0n 寄存器。

8 位定时器比较寄存器 1n（CMP1n）用于控制定时器输出（TOHn）的占空比。在定时器操作期间，可以重写 CMP1n 寄存器。

PWM 输出模式的操作如下所示。

当定时器计数启动之后，当 8 位定时器 Hn 和 CMP0n 寄存器匹配时，PWM 输出（TOHn 输出）有效电平，并将 8 位定时器 Hn 清除为 0。当 8 位定时器 Hn 和 CMP1n 寄存器匹配时，PWM 输出（TOHn 输出）非有效电平。

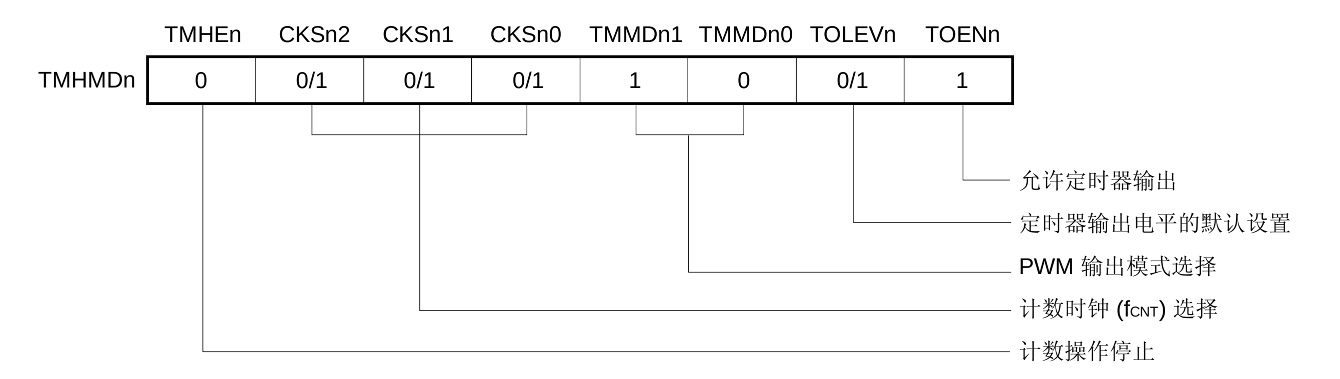
TMH2 的定时器输出（PWM 输出）只能用作 TM52 外部事件输入使能信号。请注意，不为外部输出提供引脚。

设置

<1> 设置各个寄存器。

图 8-13. PWM 输出模式下的寄存器设置

（i） 设置定时器 H 模式寄存器 n （TMHMDn）



（ii） 设置寄存器 CMP0n

- 比较值（N）： 周期设置

（iii） 设置寄存器 CMP1n

- 比较值（M）： 占空比设置

备注 **1.** n = 0 至 2。但是，TOHn 只有 TOH0 和 TOH1。
2. 00H ≤ CMP1n（M） < CMP0n（N） ≤ FFH

<2> 当 TMHEn = 1 时，计数操作开始。

<3> CMP0n 是在使能计数操作后第一个被比较的比较寄存器。当 8 位定时器计数器 Hn 的值与 CMP0n 寄存器匹配时，8 位定时器计数器 Hn 被清除，产生中断请求信号（INTTMHn），并且输出有效电平。同时，与 8 位定时器 Hn 比较的寄存器由 CMP0n 寄存器改变为 CMP1n 寄存器。

<4> 当 8 位定时器计数器 Hn 与 CMP1n 寄存器匹配时，输出非有效电平，与 8 位定时器 Hn 比较的寄存器由 CMP1n 寄存器改变为 CMP0n 寄存器。此时，8 位定时器计数器 Hn 不被清除，也不产生 INTTMHn 信号。

<5> 重复执行过程<3> 和 <4>步骤，可以获取任意占空比的脉冲。

<6> 若要停止计数操作，则设置 TMHEn = 0。

如果 CMP0n 的设置值为 N、CMP1n 的设置值为 M，计数时钟频率为 f_{CNT}，则 PWM 脉冲的输出周期和占空比如下所示。

- PWM 脉冲输出周期 = （N + 1）/f_{CNT}
- 占空比 = （M + 1）/（N + 1）

注意事项 **1. 在定时器计数操作时可以改变 CMP1n 寄存器的设置值。但是，从改变 CMP1n 寄存器的值到该值被传送到该寄存器需要经过三个操作时钟（通过 TMHMDn 寄存器的 CKSn2 至 CKSn0 位选择的时钟信号）的时间。**

2. 当停止定时器计数操作（TMHEn = 0）之后，启动定时器计数操作（TMHEn = 1）时，必须设置 CMP1n（即使将相同的值设置到 CMP1n，也必须再次设置）。

3. CMP1n 寄存器的设置值（M）和 CMP0n 寄存器的设置值（N）必须在以下范围内。

00H ≤ CMP1n（M） < CMP0n（N） ≤ FFH

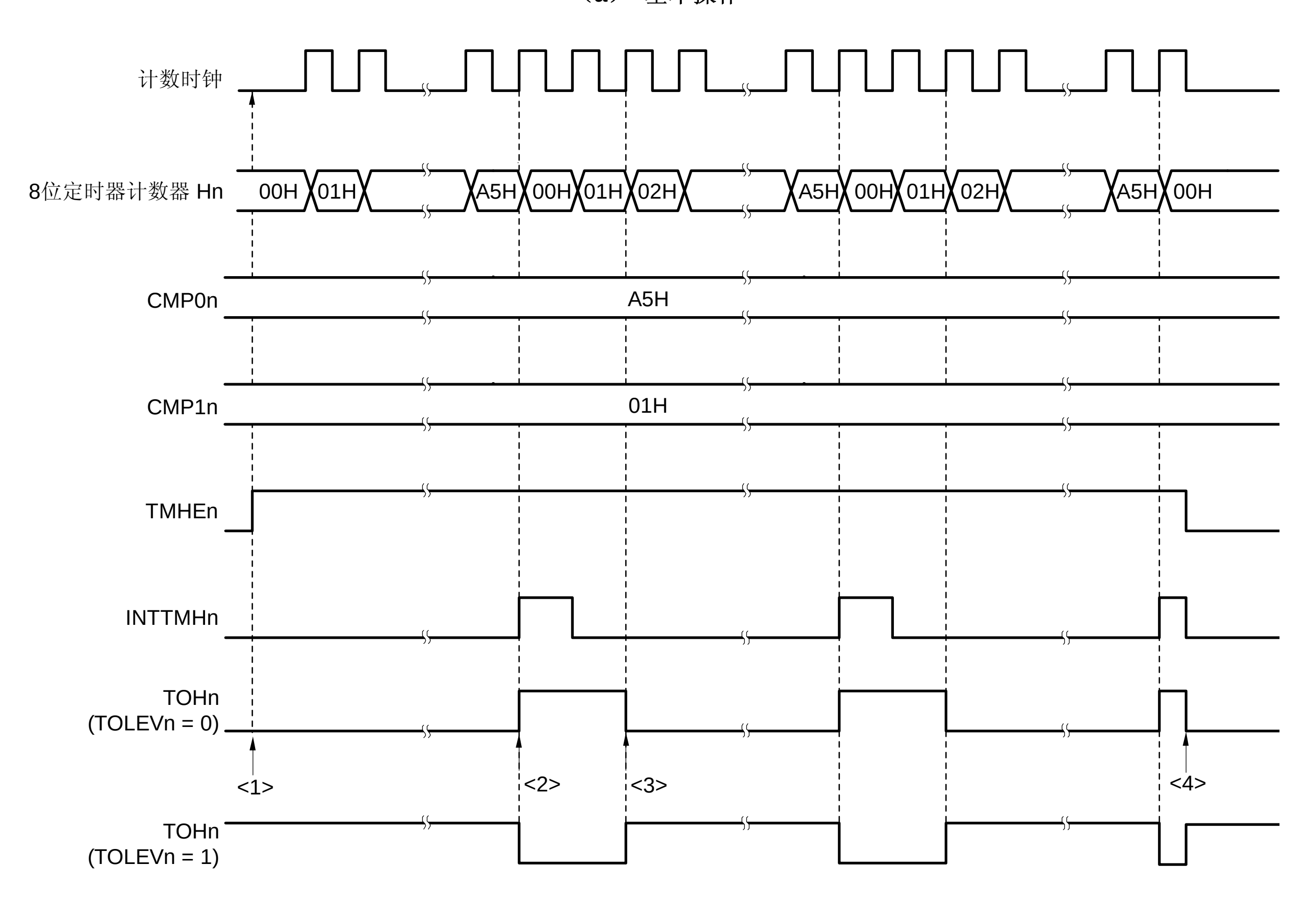
备注 **1.** 如需了解输出引脚的设置，参见 **8.3 （3） 端口模式寄存器 3（PM3）**。

2. 如需了解如何使能 INTTMHn 信号中断，参见 **第二十章 中断功能**。

3. n = 0 至 2。但是，TOHn 只有 TOH0 和 TOH1。

图 8-14. PWM 输出模式的操作时序（1/4）

（a） 基本操作



<1> 设置 TMHEn 位为 1，使能计数操作。在下一个计数时钟，启动 8 位定时器计数器 Hn 向上计数。此时，PWM 输出一个非有效电平。

<2> 当 8 位定时器计数器 Hn 的值与 CMP0n 匹配时，输出有效电平。此时，8 位定时器计数器 Hn 的值被清除，并输出 INTTMHn 信号。

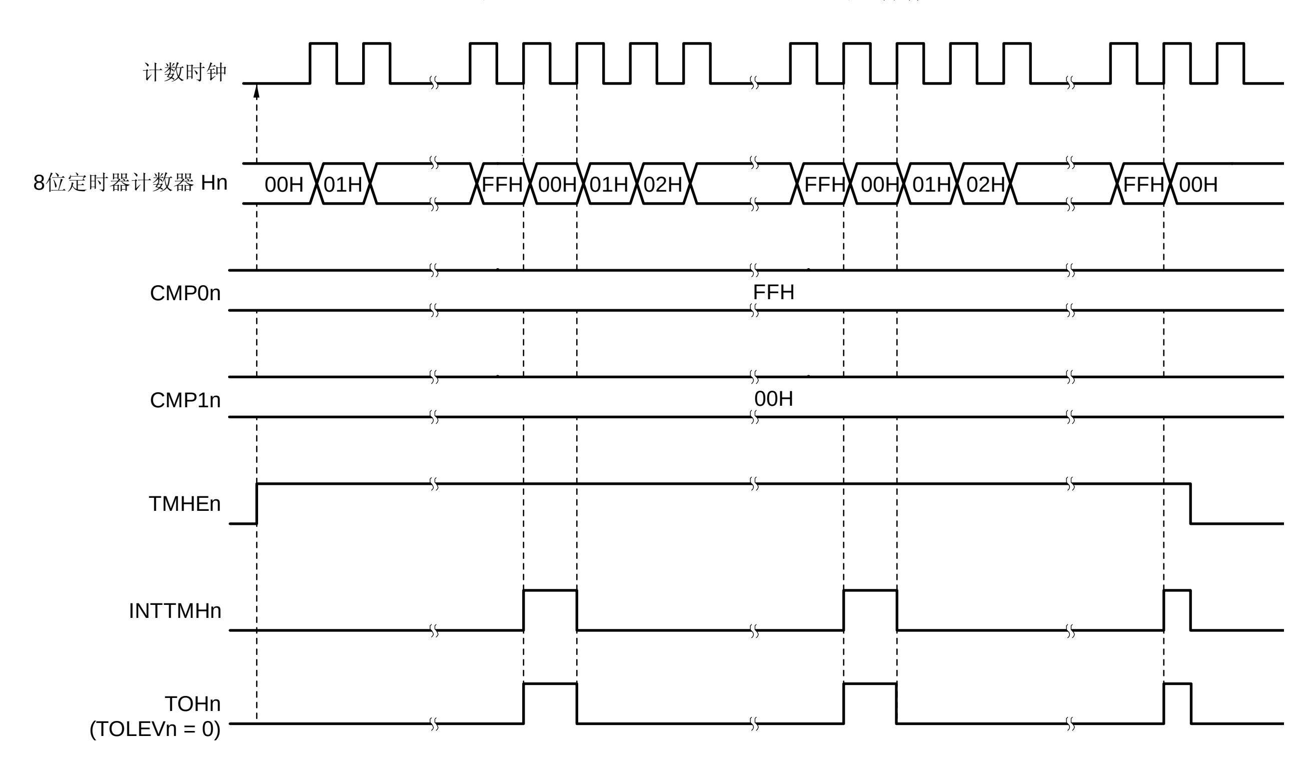
<3> 当 8 位定时器计数器 Hn 的值与 CMP1n 匹配时，输出非有效电平。此时，8 位定时器计数器 Hn 的值不被清除，也不输出 INTTMHn 信号。

<4> 在定时器 Hn 操作期间将 TMHEn 位清除为 0，可以设置 INTTMHn 信号为默认值且 PWM 输出非有效电平。

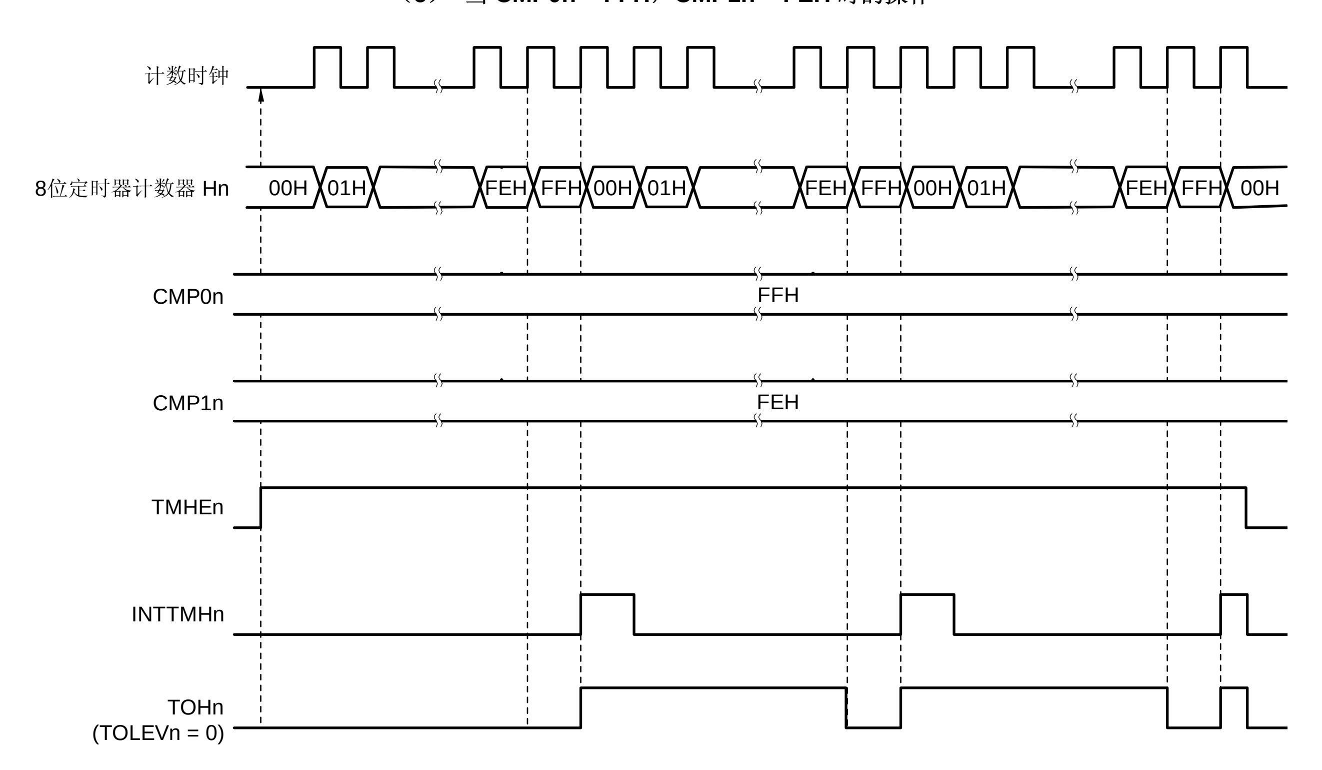
备注 n = 0 至 2。但是，TOHn 只有 TOH0 和 TOH1。

图 8-14. PWM 输出模式的操作时序（2/4）

（b） 当 CMP0n = FFH，CMP1n = 00H 时的操作



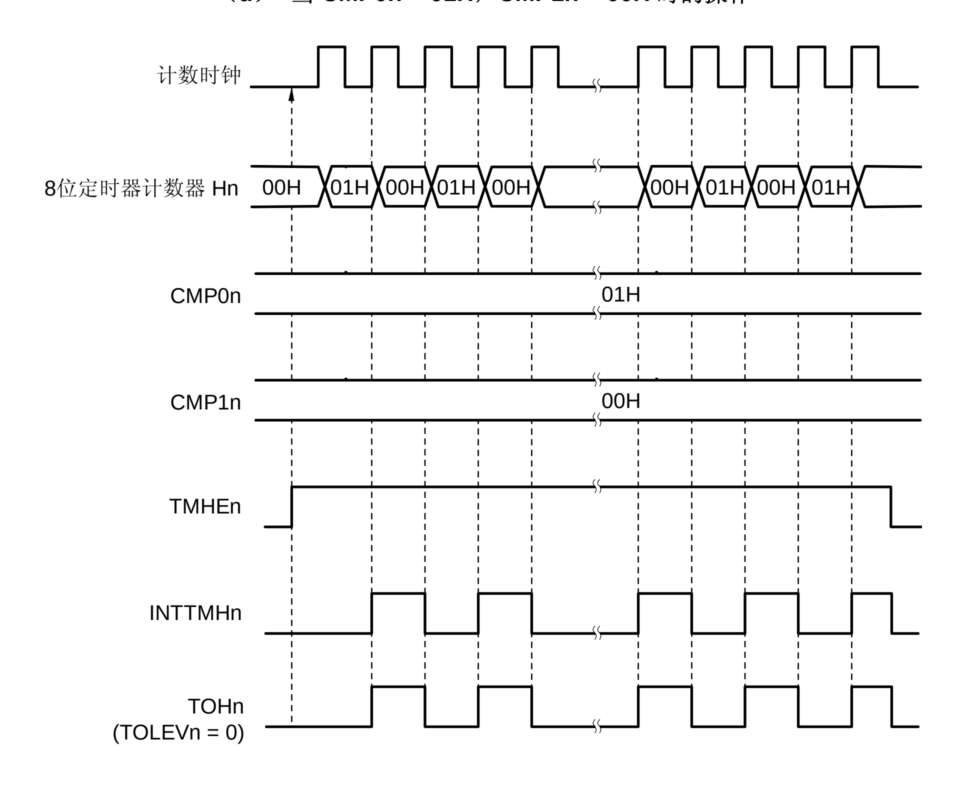
（c） 当 CMP0n = FFH，CMP1n = FEH 时的操作



备注 n = 0 至 2。但是，TOHn 只有 TOH0 和 TOH1。

图 8-14. PWM 输出模式的操作时序（3/4）

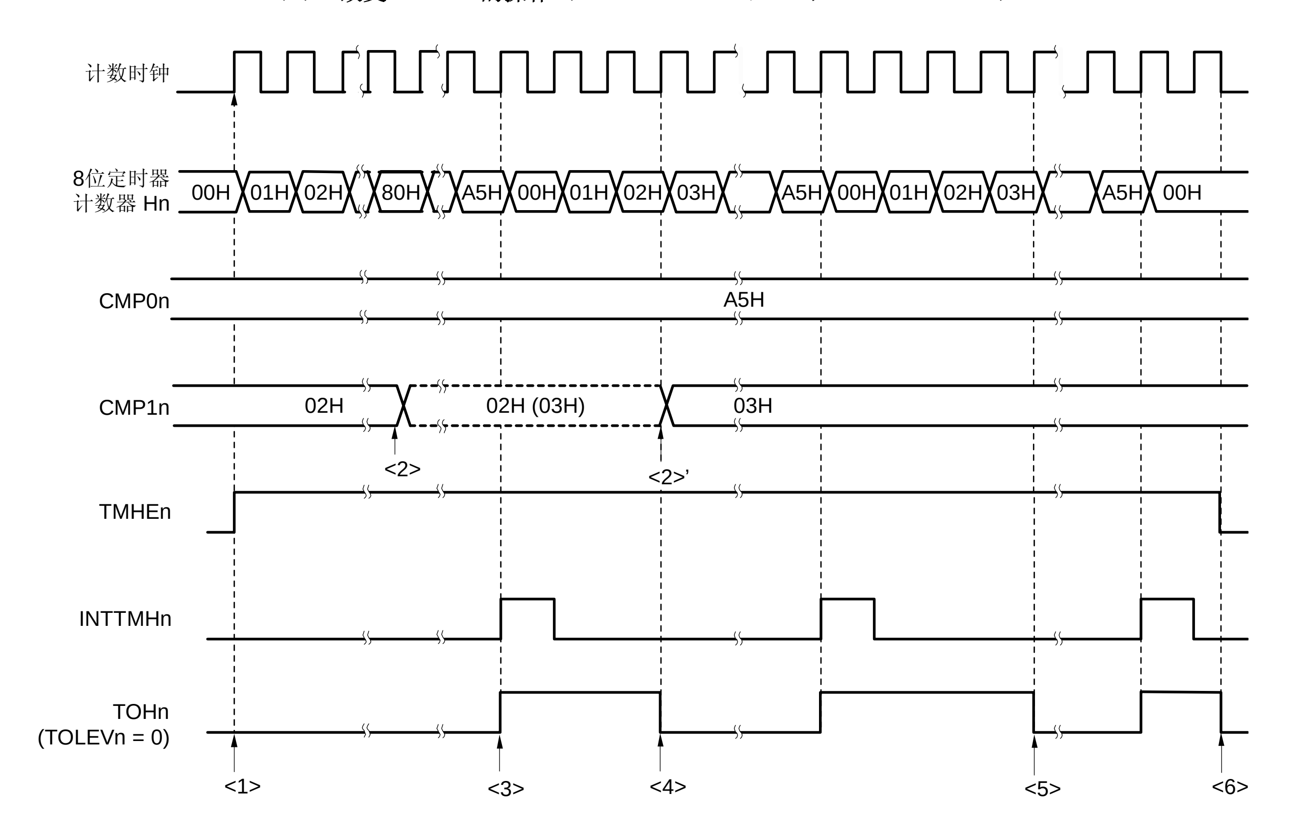
（d） 当 CMP0n = 01H，CMP1n = 00H 时的操作



备注　　n = 0 至 2。但是，TOHn 只有 TOH0 和 TOH1。

图 8-14. PWM 输出模式的操作时序（4/4）

（e） 改变 CMP1n 的操作（CMP1n = 02H → 03H， CMP0n = A5H）



<1> 设置 TMHEn 位为 1，使能计数操作。在下一个计数时钟，启动 8 位定时器计数器 Hn 向上计数。此时，PWM 输出一个非有效电平。

<2> 在定时器计数器操作期间，可以修改 CMP1n 寄存器的值。该操作与计数时钟不同步。

<3> 当 8 位定时器计数器 Hn 的值与 CMP0n 匹配时，8 位定时器计数器 Hn 的值被清除，输出有效电平，并输出 INTTMHn 信号。

<4> 如果 CMP1n 的值被修改，则修改后的值被锁存，且不传送到该寄存器。当 8 位定时器计数器 Hn 的值与 CMP1n 寄存器修改前的值匹配时，则将修改值传送到 CMP1n 寄存器，这样 CMP1n 寄存器的值被修改（<2>’）。

但是，从修改 CMP1n 的值到将该值传送到寄存器至少需要 3 个计数时钟。如果在 3 个计数时钟内产生一个匹配信号，则修改后的值不能被传送到寄存器。

<5> 当 8 位定时器计数器 Hn 的值与修改后的 CMP1n 寄存器匹配时，输出非有效电平。此时，8 位定时器计数器 Hn 的值不被清除，也不输出 INTTMHn 信号。

<6> 在定时器 Hn 操作期间将 TMHEn 位清除为 0，可以设置 INTTMHn 信号为默认值，且 PWM 输出非有效电平。

备注 n = 0 至 2。但是，TOHn 只有 TOH0 和 TOH1。

8.4.3 载波发生器操作（仅限 8 位定时器H1）

在载波发生器模式下，8 位定时器 H1 用来产生红外遥控器的载波信号，8 位定时器/事件计数器 51 用于产生红外遥控信号（计时）。

由 8 位定时器 H1 产生的载波时钟被输出。其中周期由 8 位定时器/事件计数器 51 设置。

在载波发生器模式下，8 位定时器 H1 载波脉冲的输出由 8 位定时器/事件计数器 51 控制，载波脉冲从 TOH1 输出。

（1） 载波的发生

在载波发生器模式下，8 位定时器 H 比较寄存器 01（CMP01）产生低电平宽度的载波脉冲波形，而 8 位定时器 H 比较寄存器 11（CMP11）产生高电平宽度的载波脉冲波形。

在 8 位定时器 H1 操作期间，可以重写 CMP11，但禁止重写 CMP01。

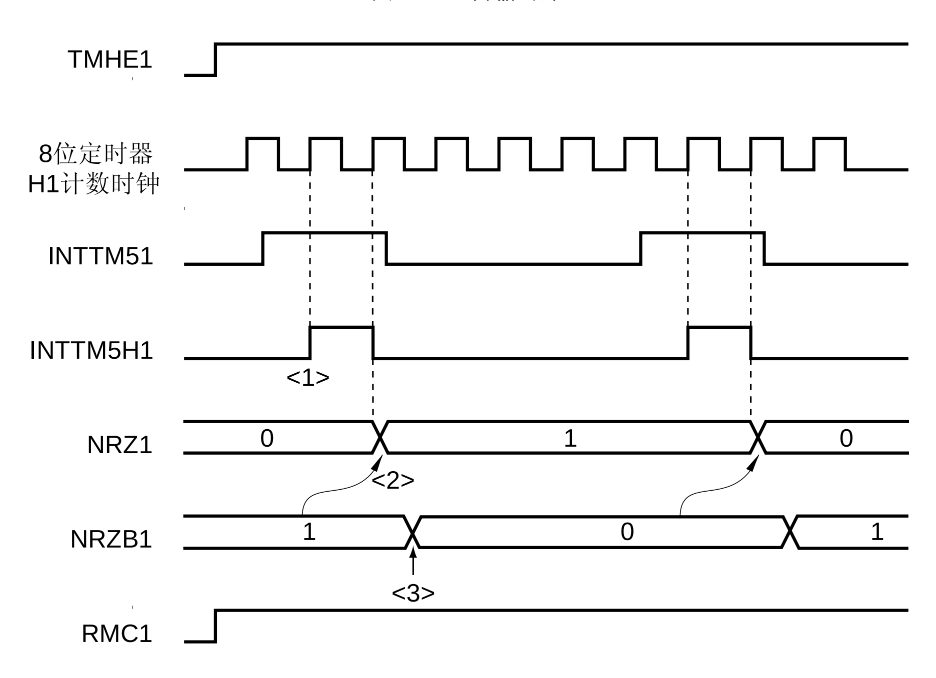
（2） 载波输出控制

载波输出由 8 位定时器/事件计数器 51 的中断请求信号（INTTM51）和 8 位定时器 H 载波控制寄存器（TMCYC1）的 NRZB1 和 RMC1 位共同控制。输出之间的关系如下所示。

RMC1 位	NRZB1 位	输出
0	0	低电平输出
0	1	在 INTTM51 信号输入的上升沿输出高电平
1	0	低电平输出
1	1	在 INTTM51 信号输入的上升沿输出载波脉冲

在计数操作期间，要控制载波脉冲输出，TMCYC1 寄存器的 NRZ1 和 NRZB1 位有 1 个主位和从位的配置。NRZ1 位是只读的，但 NRZB1 位可读可写。INTTM51 信号随着 8 位定时器 H1 计数时钟同步被输出，和 INTTM5H1 信号相同。INTTM5H1 信号成为 NRZ1 位的数据传输信号，且 NRZB1 位的值被传送到 NRZ1 位。从 NRZB1 位传送到 NRZ1 位的时序如下所示。

图 8-15. 传输时序



<1> INTTM51 信号随着 8 位定时器 H1 计数时钟同步被输出，和 INTTM5H1 信号相同。

<2> 在 INTTM5H1 信号上升沿后的第 2 个时钟时，将 NRZB1 位的值传送至 NRZ1 位。

<3> 在中断服务程序中将下一个值写入 NRZB1 位。该中断服务程序由 INTTM5H1 中断启动，或通过轮询中断请求标志检查到时序后启动。将下次要计数的值写入 CR51 中。

注意事项 1. NRZB1 位被重写后，至少要到第 2 个时钟才可以被再次重写，否则，从 NRZB1 位到 NRZ1 位的传送无法保证。

2. 在载波发生器模式下使用 8 位定时器/事件计数器 51 时，在<1>的时序处产生中断。如果在其他模式下使用 8 位定时器/事件计数器 51，中断产生的时序不同。

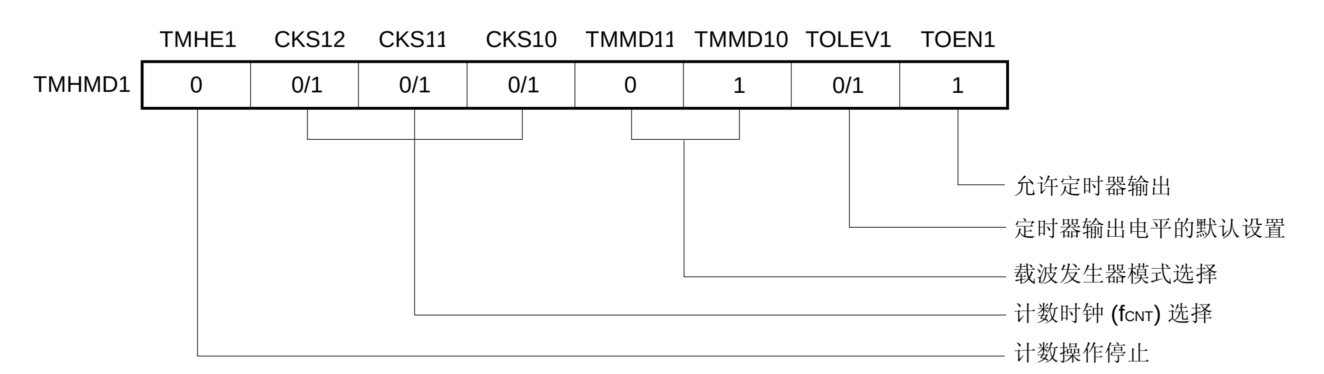
备注 INTTM5H1 是内部信号，并不是中断源。

设置

<1> 设置各个寄存器

图 8-16. 载波发生器模式的寄存器设置

（i） 设置 8 位定时器 H 模式寄存器 1 （TMHMD1）



（ii） CMP01 寄存器设置

- 比较值

（iii） CMP11 寄存器设置

- 比较值

（iv） TMCYC1 寄存器设置

- RMC1 = 1 ... 遥控输出使能位
- NRZB1 = 0/1 ... 载波输出使能位

（v） TCL51 和 TMC51 寄存器设置

- 参见 **7.3 控制 8 位定时器/事件计数器 50、51 和 52 的寄存器**。

<2> 当 TMHE1 = 1 时，8 位定时器 H1 开始计数。

<3> 当 8 位定时器模式控制寄存器 51（TMC51）的 TCE51 被设置为 1 时，8 位定时器/事件计数器 51 开始计数。

<4> 计数操作使能之后，首个用于比较的比较寄存器是 CMP01。当 8 位定时器计数器 H1 的计数值与 CMP01 的值匹配时，产生 INTTMH1 信号，8 位定时器计数器 H1 被清除。同时，与 8 位定时器计数器 H1 比较的比较寄存器由 CMP01 寄存器切换为 CMP11 寄存器。

<5> 当 8 位定时器计数器 H1 的计数值与 CMP11 寄存器的值匹配时，产生 INTTMH1 信号，8 位定时器计数器 H1 被清除。同时，与 8 位定时器计数器比较的比较寄存器由 CMP11 寄存器切换为 CMP01 寄存器。

<6> 重复执行过程<4>和<5>步骤，将产生 1 个载波时钟。

<7> INTTM51 信号随着 8 位定时器 H1 计数时钟同步被输出，和 INTTM5H1 信号相同。INTTM5H1 信号成为 NRZB1 位的数据传输信号，并将 NRZB1 位的值传送到 NRZ1 位。

<8> 在中断服务程序中将下一个值写入 NRZB1 位。该中断服务程序已由 INTTM5H1 中断启动，或通过轮询中断请求标志检查到时序后启动。将下次要计数的值写入 CR51 中。

<9> 当 NRZ1 位是高电平时，载波时钟将从 TOH1 输出。

<10> 通过执行上述过程，可以获得任意频率的载波时钟。若要停止计数操作，将 TMHE1 清除为 0。

如果 CMP01 的设置值为 N、CMP11 的设置值为 M、计数时钟频率为 f_{CNT}，则载波时钟输出周期与占空比如下所示。

- 载波时钟输出周期 = （N + M + 2）/f_{CNT}
- 占空比=高电平宽度/ 载波时钟输出宽度 = （ M + 1） / （N + M + 2）

注意事项 **1. 当停止定时器计数操作（TMHE1 = 0）之后，启动定时器计数操作（TMHE1 = 1）时，必须设置 CMP11（即使将相同的值设置到 CMP11，也必须再次设置）。**

2. TMH1 的计数时钟频率要设置为 TM51 计数时钟频率的 6 倍以上。

3. CMP01 和 CMP11 的设置值必须在 01H 和 FFH 范围内。

4. 在定时器计数操作时可以改变 CMP11 寄存器的设置值。但是，从改变 CMP11 寄存器的值到该值被传送到该寄存器需要经过三个操作时钟（通过 TMHMD1 寄存器的 CKS12 至 CKS10 位选择的时钟信号）的时间。

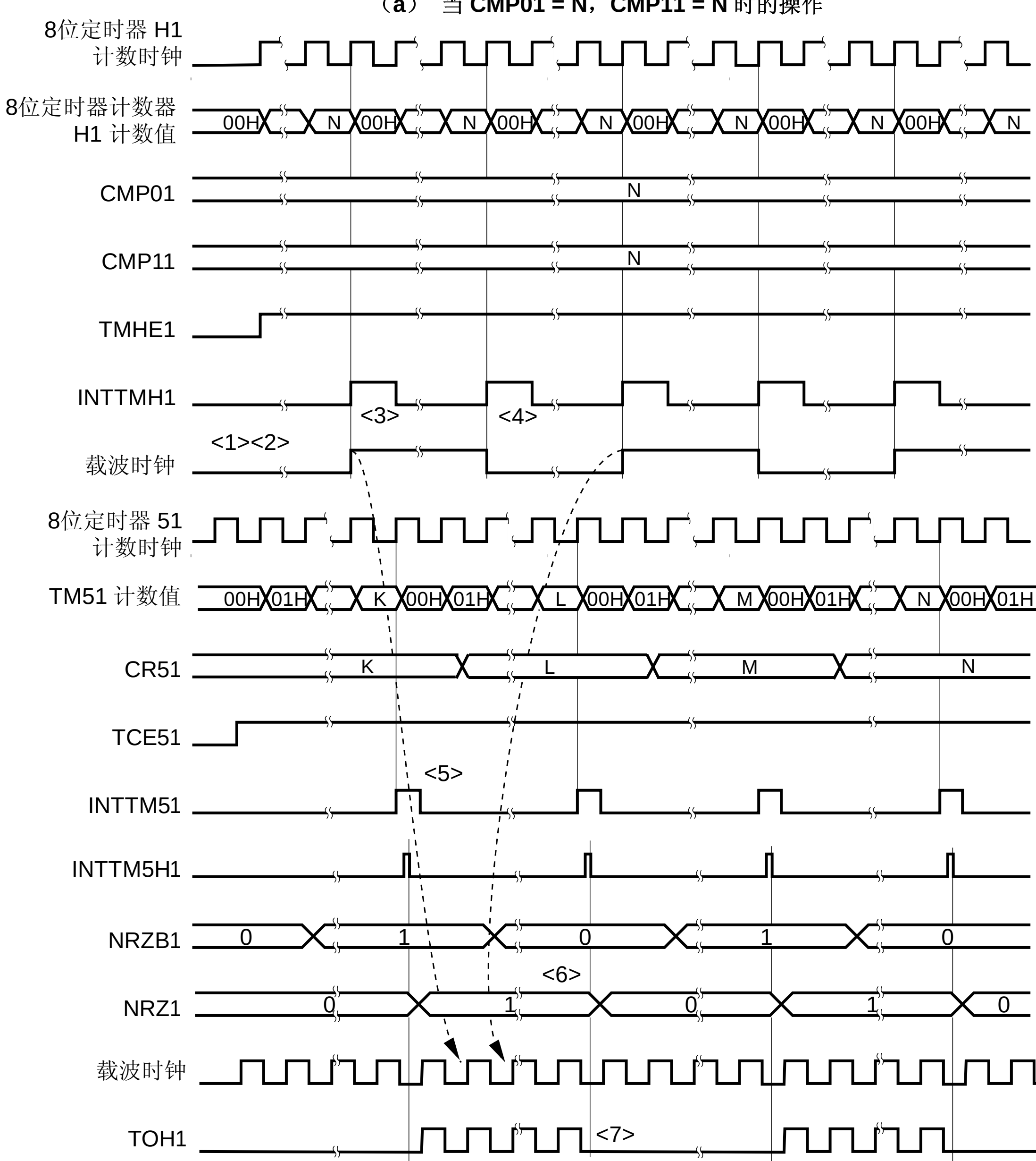
5. 在计时操作开始前，必须设置 RMC1 位。

备注 **1.** 如需了解输出引脚的设置，参见 **8.3 （3） 端口模式寄存器 3（PM3）**。

2. 如需了解如何使能 INTTMH1 信号中断，参见 第二十章 中断功能。

图 8-17. 载波发生器模式的操作时序（1/3）

（a） 当 CMP01 = N，CMP11 = N 时的操作



<1> 当 TMHE1 = 0 且 TCE51 = 0 时，停止 8 位定时器计数器 H1 的操作。

<2> 当 TMHE1 = 1 时，8 位定时器计数器 H1 开始计数操作。此时，载波时钟保持默认状态。

<3> 当 8 位定时器计数器 H1 的计数值与 CMP01 寄存器的值匹配时，产生第 1 个 INTTMH1 信号，并反转载波时钟信号，与 8 位定时器计数器 H1 比较的比较寄存器从 CMP01 寄存器切换为 CMP11 寄存器。8 位定时器计数器 H1 被清除为 00H。

<4> 当 8 位定时器计数器 H1 的计数值与 CMP11 寄存器的值匹配时，产生 INTTMH1 信号，并反转载波时钟信号，与 8 位定时器计数器 H1 比较的比较寄存器从 CMP11 寄存器切换为 CMP01 寄存器。8 位定时器计数器 H1 被清除为 00H。重复执行过程<3>和<4>步骤，将产生 1 个载波时钟，且占空比固定为 50%。

<5> 当产生 INTTM51 信号时，它随着 8 位定时器 H1 计数时钟同步被输出，和 INTTM5H1 信号相同。

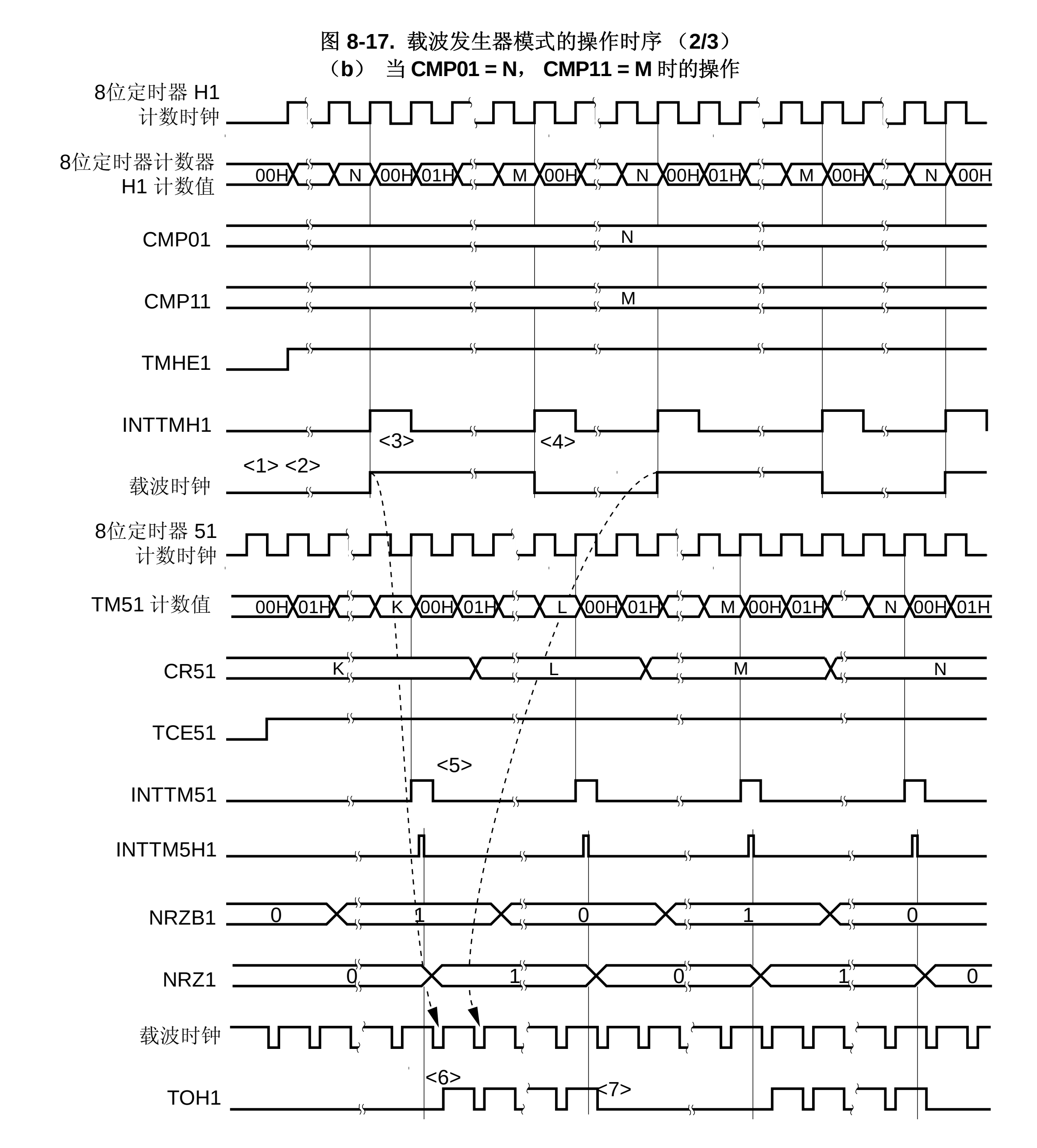
<6> INTTM5H1 信号成为 NRZB1 位的数据传输信号，并将 NRZB1 位的值传送到 NRZ1 位。

<7> 当设置 NRZ1 = 0 时，TOH1 输出变为低电平。

备注　INTTM5H1 是内部信号，并不是中断源。

图 8-17. 载波发生器模式的操作时序 （2/3）

（b） 当 CMP01 = N， CMP11 = M 时的操作



<1> 当 TMHE1 = 0 且 TCE51 = 0 时，停止 8 位定时器计数器 H1 的操作。

<2> 当 TMHE1 = 1 时，8 位定时器计数器 H1 开始计数操作。此时，载波时钟保持默认状态。

<3> 当 8 位定时器计数器 H1 的计数值与 CMP01 寄存器的值匹配时，产生第 1 个 INTTMH1 信号，并反转载波时钟信号，与 8 位定时器计数器 H1 比较的比较寄存器从 CMP01 寄存器切换为 CMP11 寄存器。8 位定时器计数器 H1 被清除为 00H。

<4> 当 8 位定时器计数器 H1 的计数值与 CMP11 寄存器的值匹配时，产生 INTTMH1 信号，并反转载波时钟信号，与 8 位定时器计数器 H1 比较的比较寄存器从 CMP11 寄存器切换为 CMP01 寄存器。8 位定时器计数器 H1 被清除为 00H。重复执行过程<3>和<4>步骤，将产生 1 个载波时钟，且占空比固定为 50%以外的值。

<5> 当产生 INTTM51 信号时，它随着 8 位定时器 H1 计数时钟同步被输出，和 INTTM5H1 信号相同。

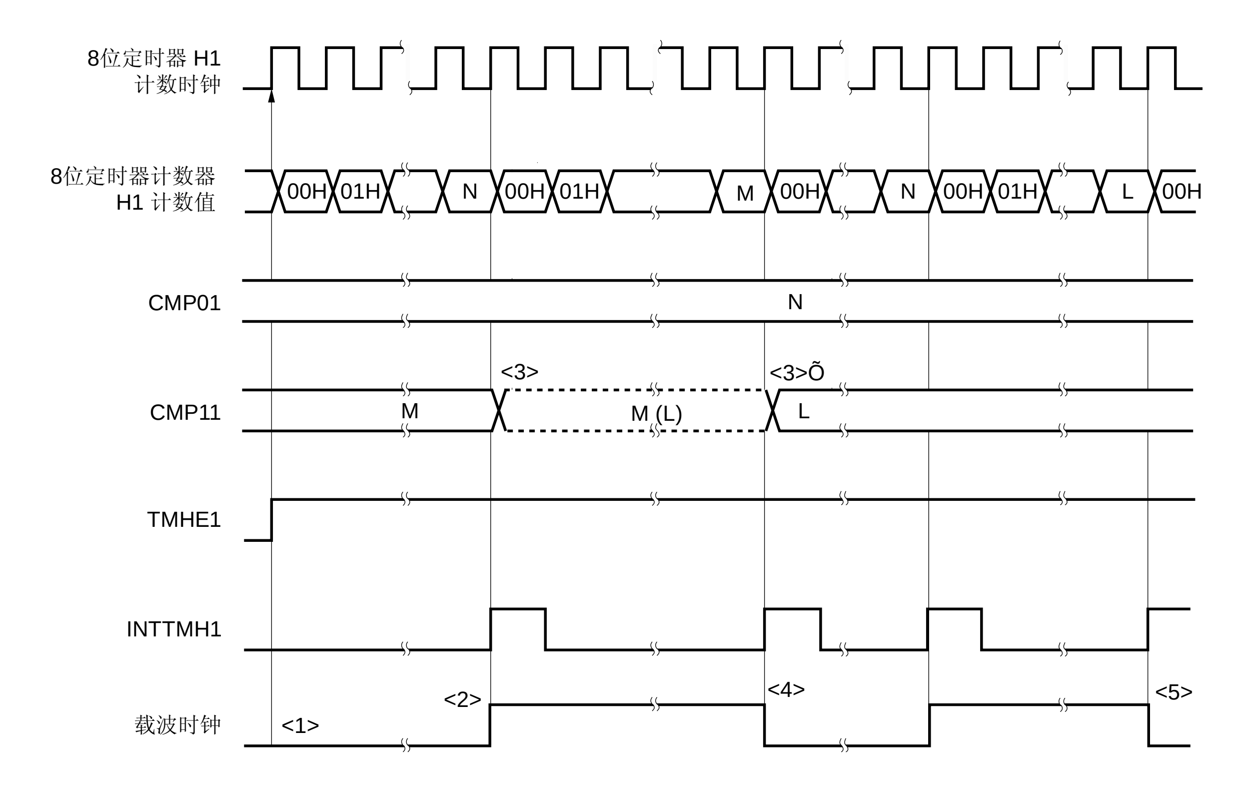
<6> 如果 NRZ1 被设置为 1，在载波时钟的第 1 个上升沿处输出一个载波信号。

<7> 当 NRZ1 = 0 时，TOH1 输出保持为高电平，当载波时钟为高电平时（在<6>和<7>步骤中，保证载波时钟波形的高电平宽度）也不会改变为低电平。

备注 INTTM5H1 是内部信号，并不是中断源。

图 8-17. 载波发生器模式的操作时序 （3/3）

（c） 当 CMP11 改变时的操作



<1> 当 TMHE1 = 1 时，8 位定时器 H1 开始计数操作。此时，载波时钟保持默认状态。

<2> 当 8 位定时器计数器 H1 的计数值与 CMP01 的值匹配时，输出 INTTMH1 信号、反转载波信号，定时器计数器被清除为 00H。同时，用于将其值与 8 位定时器计数器 H1 比较的比较寄存器从 CMP01 寄存器切换为 CMP11 寄存器。

<3> CMP11 寄存器与计数时钟不同步，在 8 位定时器 H1 操作期间，可以修改它的值。修改后的寄存器的新值（L）被锁存。当 8 位定时器计数器 H1 的计数值与 CMP11 寄存器修改前的值（M）匹配时，CMP11 寄存器的值被修改（<3>’）。

但是，从修改 CMP11 的值到将该值传送到寄存器至少需要 3 个计数时钟。如果在 3 个计数时钟内产生一个匹配信号，则修改后的值不能被传送到寄存器。

<4> 当 8 位定时器计数器 H1 的计数值与 CMP11 修改前的值（M）匹配时，输出 INTTMH1 信号、反转载波信号，并将定时器计数器清除为 00H。同时，于将其值与 8 位定时器计数器 H1 比较的比较寄存器从 CMP11 寄存器切换为 CMP01 寄存器。

<5> 由修改后的值（L）确定 8 位定时器计数器 H1 的计数值与 CMP11 的值再次匹配时的时序。

<R>

8.4.4 由定时器 51 计数器控制载波时钟的数量

要控制从 TOH1 引脚输出的载波时钟的数量，可以将定时器 H1 的输出信号选择作为 8 位定时器 51 计数时钟。

图 8-18 展示了从 TOH1 引脚输出 3 个载波时钟的控制示例。

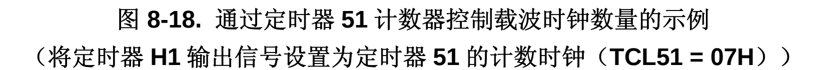


图 8-18. 通过定时器 51 计数器控制载波时钟数量的示例

（将定时器 H1 输出信号设置为定时器 51 的计数时钟（TCL51 = 07H））

<1> 要从 TOH1 引脚输出 3 个载波时钟时，设置 CR51 寄存器为 02H。

<2> 当 TM51 计数值和 CR51 寄存器的值（02H）匹配时，产生 INTTM51 信号。该信号随着 8 位定时器 H1 计数时钟同步被输出，和 INTTM5H1 信号相同。

<6> INTTM5H1 信号成为 NRZB1 位的数据传输信号，并将 NRZB1 位的值传送到 NRZ1 位。
此时的传输时序是在 INTTM5H1 信号上升沿后定时器 H1 的第 2 个计数时钟处。

<7> 当设置 NRZ1 为 0 时，在已经输出第三个载波时钟后，TOH1 输出变为低电平。

备注 INTTM5H1 是内部信号，并不是中断源。

第九章 实时计数器

9.1 实时计数器的功能

实时计数器具有如下特性。

- 具有年、月、星期、日、小时、分钟和秒的计数器，可以计数到 99 年。
- 固定周期中断功能（周期：1 个月到 0.5 秒）
- 闹钟中断功能（闹钟：星期、小时、分钟）
- 间隔中断功能
- 1 Hz 的引脚输出功能
- 512 Hz 或 16.384 kHz 或 32.768 kHz 的引脚输出功能

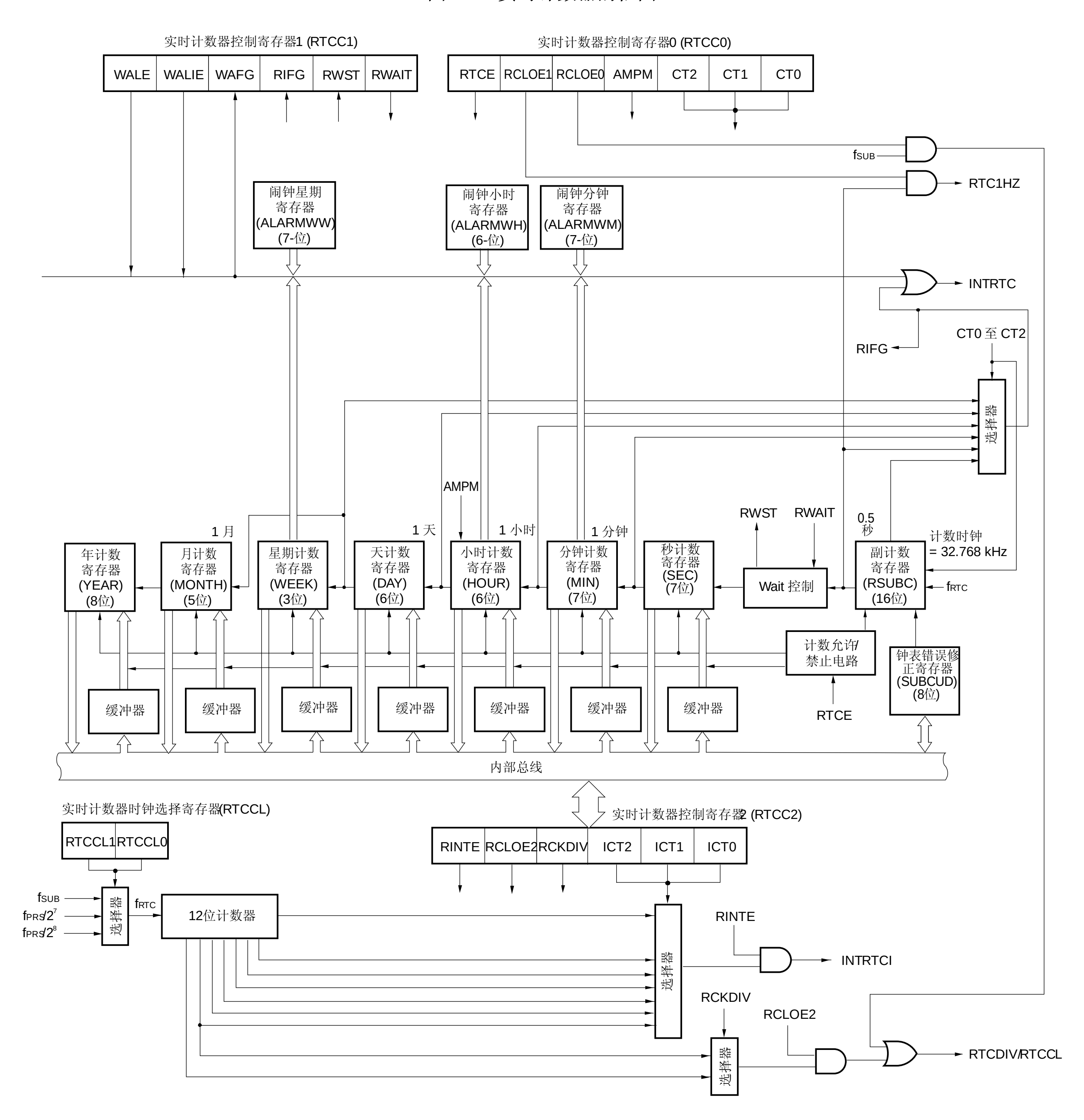
9.2 实时计数器的配置

实时计数器包括以下硬件。

表 9-1. 实时计数器的配置

项目	配置
控制寄存器	实时计数器时钟选择寄存器（RTCCL）
	实时计数器控制寄存器 0（RTCC0）
	实时计数器控制寄存器 1（RTCC1）
	实时计数器控制寄存器 2（RTCC2）
	副计数寄存器（RSUBC）
	秒计数寄存器（SEC）
	分钟计数寄存器（MIN）
	小时计数寄存器（HOUR）
	日计数寄存器（DAY）
	星期计数寄存器（WEEK）
	月计数寄存器（MONTH）
	年计数寄存器（YEAR）
	钟表错误修正寄存器（SUBCUD）
	闹钟分钟寄存器（ALARMWM）
	闹钟小时寄存器（ALARMWH）
	闹钟星期寄存器（ALARMWW）

图 9-1. 实时计数器的框图



9.3 控制实时计数器的寄存器

以下 16 个寄存器用于控制实时计数器。

（1）实时计数器时钟选择寄存器（RTCCL）

该寄存器用于控制实时计数器的模式。

可以通过 1 位或 8 位存储器操作指令来设置 RTCCL。

复位信号的产生会将该寄存器清除为 00H。

图 9-2. 实时计数器时钟选择寄存器（RTCCL）的格式

地址：FF54H　复位后：00H　R/W

符号	7	6	5	4	3	2	<1>	<0>
RTCCL	0	0	0	0	0	0	RTCCL1	RTCCL0

RTCCL1	RTCCL0	实时计数器（RTC）输入时钟（f_{RTC}）的控制
0	0	f_{SUB}
0	1	$f_{PRS}/2^7$
1	0	$f_{PRS}/2^8$
1	1	禁止设置

备注
- 当 f_{PRS} = 4.19 MHz，$f_{RTC} = f_{PRS}/2^7$ = 32.768 kHz
- 当 f_{PRS} = 8.38 MHz，$f_{RTC} = f_{PRS}/2^8$ = 32.768 kHz

（2）实时计数器控制寄存器 0（RTCC0）

RTCC0 寄存器是 8 位寄存器，用于启动或停止实时计数器操作、控制 RTCCL 和 RTC1HZ 引脚、设置 12 或 24 小时系统和固定周期中断功能。

可以通过 1 位或 8 位存储器操作指令来设置 RTCC0。

复位信号的产生会将该寄存器清除为 00H。

图 9-3. 实时计数器控制寄存器 0（RTCC0）的格式

地址：FF89H 复位后：00H R/W

符号	<7>	6	<5>	<4>	3	2	1	0
RTCC0	RTCE	0	RCLOE1	RCLOE0	AMPM	CT2	CT1	CT0

RTCE	实时计数器操作控制
0	停止计数器操作
1	启动计数器操作

RCLOE1	RTC1HZ 引脚输出控制
0	禁止 RTC1HZ 引脚的输出（1 Hz）
1	使能 RTC1HZ 引脚的输出（1 Hz）

RCLOE0[注]	RTCCL 引脚输出控制
0	禁止 RTCCL 引脚的输出（32.768 kHz）
1	使能 RTCCL 引脚的输出（32.768 kHz）

AMPM	选择12/24小时系统
0	12 小时系统（显示 a.m. 和 p.m. ）
1	24 小时系统
• 改变 AMPM 的值，将 RWAIT（RTCC1 的第 0 位）设置为 1，并重置小时计数寄存器（HOUR）。 • 表 9-2 展示显示的时间数字。	

CT2	CT1	CT0	固定周期中断（INTRTC）选择
0	0	0	不使用固定周期中断功能
0	0	1	每 0.5 s 一次（与秒向上计数同步）
0	1	0	每 1 s 一次（与秒向上计数同时）
0	1	1	每 1 m 一次（每分钟的 00 秒）
1	0	0	每 1 小时一次（每小时的 00 分钟 00 秒）
1	0	1	每 1 天一次（每天的 00 小时 00 分钟 00 秒）
1	1	×	每 1 月一次（每月的第 1 天 a.m 00 小时 00 分 00 秒）
CT2 至 CT0 的值改变之后，清除中断请求标志。			

注　不能同时使能 RCLOE0 和 RCLOE2 。

注意事项　**如果当 RTCE = 1 时改变 RCLOE0 和 RCLOE1，32.768 kHz 和 1Hz 输出信号将可能产生一个宽度较窄的脉冲。**

备注　×：无须理会

（3）实时计数器控制寄存器 1（RTCC1）

RTCC1 寄存器是 8 位寄存器，用于控制闹钟中断功能和计数器的等待时间。

可以通过 1 位或 8 位存储器操作指令来设置 RTCC1。

复位信号的产生会将该寄存器清除为 00H。

图 9-4. 实时计数器控制寄存器 1（RTCC1）的格式（1/2）

地址：FF8AH　复位后：00H　R/W

符号	<7>	<6>	5	<4>	<3>	2	<1>	<0>
RTCC1	WALE	WALIE	0	WAFG	RIFG	0	RWST	RWAIT

WALE	闹钟操作控制
0	匹配操作无效
1	匹配操作有效
设置闹钟寄存器（RTCC1 的 WALIE 位，ALARMWM 寄存器，ALARMWH 寄存器和 ALARMWW 寄存器），禁止 WALE（将其清除为“0”）。	

WALIE	闹钟中断（INTRTC）功能操作的控制
0	闹钟匹配时不产生中断。
1	闹钟匹配时产生中断。

WAFG	闹钟检测状态标志
0	闹钟不匹配
1	检测闹钟匹配
该状态标志表示闹钟匹配的检测。仅当 WALE = 1 时有效，且在检测到闹钟匹配的一个时钟（32.768 kHz）后被设置为“1”。向该标志写入“0”会清除该标志，写入“1” 无效。	

RIFG	固定周期中断状态标志
0	不产生固定周期中断。
1	产生固定周期中断。
该标志表示固定周期中断的产生状态。当产生固定周期中断时，该位置 “1”。 向该标志写入“0”会清除该标志，写入“1” 无效。	

RWST	实时计数器的等待状态标志
0	计数器操作中。
1	读取或写入计数器值模式。
该状态标志表示 RWAIT 的设置是否有效。 读取或写入计数器值之前，确认该标志的值为 1。	

图 **9-4.** 实时计数器控制寄存器 **1**（**RTCC1**）的格式（**2/2**）

RWAIT	实时计数器的等待控制
0	设置计数器操作
1	停止 SEC 至 YEAR 计数器。读取或写入计数器值模式
该位用于控制计数器的操作。 请确保写入 “1” ，来读取或写入计数器值。 因为 RSUBC 继续操作，在 1 秒内完成读取或写入，并将该位清除回 0。 当 RWAIT = 1 时，需要经过 1 个时钟（32.768 kHz），计数器值才能被读出或写入。 当 RWAIT = 1 时，如果 RSUBC 溢出，在 RWAIT = 0 之后它向上计数。如果秒计数寄存器被写入，但是，因为 RSUBC 被清除，它不会向上计数。	

<R> **注意事项** **如果使用 1 位操作指令写入 RTCC1 寄存器，则 RIFG 和 WAFG 标志可能被清除。因此，为了写入 RTCC1 寄存器，请使用 8 位操作指令。要在写入期间避免 RIFG 和 WAFG 标志被清除，向对应的位设置“1”来禁止写入。不使用 RIFG 和 WAFG 标志时，当该值可能被重写时，可以使用 1 位操作指令写入 RTCC1 寄存器。**

备注 固定周期中断和闹钟匹配中断使用相同的中断源（INTRTC）。当同时使用这两种类型的中断时，在 INTRTC 产生时检查固定周期中断状态标志（RIFG）和闹钟检测状态标志（WAFG），来判断是哪种中断发生。

（4）实时计数器控制寄存器 2（RTCC2）

RTCC2 寄存器是 8 位寄存器，它用于控制间隔中断功能和 RTCDIV 引脚。

可以通过 1 位或 8 位存储器操作指令来设置 RTCC2。

复位信号的产生会将该寄存器清除为 00H。

图 9-5. 实时计数器控制寄存器 2（RTCC2）的格式

地址：FF8BH 复位后：00H R/W

符号	<7>	<6>	<5>	4	3	2	1	0
RTCC2	RINTE	RCLOE2	RCKDIV	0	0	ICT2	ICT1	ICT0

RINTE	ICT2	ICT1	ICT0	选择间隔中断（INTRTCI）
0	×	×	×	不产生间隔中断
1	0	0	0	$2^6/f_{RTC}$（1.953125 ms）
1	0	0	1	$2^7/f_{RTC}$（3.90625 ms）
1	0	1	0	$2^8/f_{RTC}$（7.8125 ms）
1	0	1	1	$2^9/f_{RTC}$（15.625 ms）
1	1	0	0	$2^{10}/f_{RTC}$（31.25 ms）
1	1	0	1	$2^{11}/f_{RTC}$（62.5 ms）
1	1	1	×	$2^{12}/f_{RTC}$（125 ms）

RCLOE2[注]	RTCDIV 引脚输出控制
0	禁止 RTCDIV 引脚的输出。
1	使能 RTCDIV 引脚的输出。

RCKDIV	RTCDIV 引脚输出频率的选择
0	RTCDIV 引脚输出 512 Hz (1.95 ms)
1	RTCDIV 引脚输出 16.384 kHz (0.061 ms)

注 不能同时使能 RCLOE0 和 RCLOE2 。

注意事项 1. 当 RINTE = 0 时，改变 ICT2，ICT1 和 ICT0。

2. 当 RTCDIV 引脚输出停止时，在最多 2 个 f_{RTC} 时钟内输出继续且进入低电平。在输出 512 Hz 时，当进入高电平后立即停止输出，可能会产生一个脉冲，其宽度至少是 1 个 f_{XT} 时钟。

（**5**）副计数寄存器（**RSUBC**）

RSUB 寄存器是 16 位寄存器，用于计数实时计数器的 1 秒参考时间。它的值是 0000H 至 7FFFH，使用 32.768 kHz 时钟计数 1 秒钟。

可以使用 16 位存储器操作指令来设置 RSUBC。

复位信号的产生会将该寄存器清除为 0000H。

注意事项 1.　当使用 SUBCUD 寄存器修正时，该值可能会变为 8000H 或更大。
2.　通过写入秒计数寄存器的影响，产生的重置将该寄存器清除。
3.　在操作期间读取该寄存器，不能保证读出值，因为读取的值正在改变。

图 9-6. 副计数寄存器（RSUBC）的格式

地址：FF60H　复位后：0000H　R

符号	7	6	5	4	3	2	1	0
RSUBC	SUBC7	SUBC6	SUBC5	SUBC4	SUBC3	SUBC2	SUBC1	SUBC0

地址：FF61H　复位后：0000H　R

符号	7	6	5	4	3	2	1	0
RSUBC	SUBC15	SUBC14	SUBC13	SUBC12	SUBC11	SUBC10	SUBC9	SUBC8

（6）秒计数寄存器（**SEC**）

SEC 寄存器是 8 位寄存器，它的值是 0 至 59（十进制），用于表示秒的计数值。

当副计数器溢出时，它将向上计数。

向该寄存器写入数据时，数据被写入缓冲器，并在最多 2 个时钟（32.768 kHz）后写入计数器。以 BCD 码设置 0 至 59 的十进制数到该寄存器。如果设置值超出该范围，在 1 个周期后寄存器的值将返回到正常值。

可以使用 8 位存储器操作指令来设置 SEC。

复位信号的产生会将该寄存器清除为 00H。

图 9-7. 秒计数寄存器（SEC）的格式

地址：FF62H　复位后：00H　R/W

符号	7	6	5	4	3	2	1	0
SEC	0	SEC40	SEC20	SEC10	SEC8	SEC4	SEC2	SEC1

（7）分钟计数寄存器（MIN）

MIN 寄存器是 8 位寄存器，它的值是 0 至 59（十进制），用于表示分钟的计数值。

当秒计数器溢出时，它将向上计数。

向该寄存器写入数据时，数据被写入缓冲器，并在最多 2 个时钟（32.768 kHz）后写入计数器。以 BCD 码设置 0 至 59 的十进制数到该寄存器。如果设置值超出该范围，在 1 个周期后寄存器的值将返回到正常值。

可以使用 8 位存储器操作指令来设置 MIN。

复位信号的产生会将该寄存器清除为 00H。

图 9-8. 分钟计数寄存器（MIN）的格式

地址：FF63H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
MIN	0	MIN40	MIN20	MIN10	MIN8	MIN4	MIN2	MIN1

（8）小时计数寄存器（HOUR）

HOUR 寄存器是 8 位寄存器，它的值是 00 至 23 或 01 至 12，21 至 32（十进制），用于表示小时的计数值。

当分钟计数器溢出时，它将向上计数。

向该寄存器写入数据时，数据被写入缓冲器，并在最多 2 个时钟（32.768 kHz）后写入计数器。以 BCD 码设置 00 至 23 或 01 至 12、21 至 32 的十进制数到该寄存器。如果设置值超出该范围，在 1 个周期后寄存器的值将返回到正常值。

可以使用 8 位存储器操作指令来设置 HOUR。

复位信号的产生会将该寄存器清除为 12H。

但是，如果复位后 AMPM 位被置为 1，该寄存器的值是 00H。

图 9-9. 小时计数寄存器（HOUR）的格式

地址：FF64H 复位后：12H R/W

符号	7	6	5	4	3	2	1	0
HOUR	0	0	HOUR20	HOUR10	HOUR8	HOUR4	HOUR2	HOUR1

注意事项 **当 AMPM = 0（如果选择 12 小时系统）时，HOUR 的第 5 位（HOUR20）表示 AM（0）/PM（1）。**

表 9-2 展示了 AMPM 位设置值、HOUR 寄存器值和时间之间的关系

<R>

表 9-2. 显示的时间数字

24 小时显示 (AMPM 位 = 1)		12 小时显示 (AMPM 位 = 0)	
时间	HOUR 寄存器	时间	HOUR 寄存器
0	00H	0 a.m.	12H
1	01H	1 a.m.	01H
2	02H	2 a.m.	02H
3	03H	3 a.m.	03H
4	04H	4 a.m.	04H
5	05H	5 a.m.	05H
6	06H	6 a.m.	06H
7	07H	7 a.m.	07H
8	08H	8 a.m.	08H
9	09H	9 a.m.	09H
10	10H	10 a.m.	10H
11	11H	11 a.m.	11H
12	12H	0 p.m.	32H
13	13H	1 p.m.	21H
14	14H	2 p.m.	22H
15	15H	3 p.m.	23H
16	16H	4 p.m.	24H
17	17H	5 p.m.	25H
18	18H	6 p.m.	26H
19	19H	7 p.m.	27H
20	20H	8 p.m.	28H
21	21H	9 p.m.	29H
22	22H	10 p.m.	30H
23	23H	11 p.m.	31H

当 AMPM 位是“0”时，HOUR 寄存器的值被设置为 12 小时显示，当 AMPM 位是“1”时，被设置为 24 小时显示。在 12 小时显示时，HOUR 寄存器的第 5 位在 AM 时为显示 0，在 PM 时显示 1。

（9）日计数寄存器（DAY）

DAY 寄存器是 8 位寄存器，它的值是 1 至 31（十进制），用于表示日的计数值。
当小时计数器溢出时，它将向上计数。
该计数器计数方式如下。

- 01 至 31（一月、三月、五月、七月、八月、十月、十二月）
- 01 至 30（四月、六月、九月、十一月）
- 01 至 29（二月、闰年）
- 01 至 28（二月、平年）

向该寄存器写入数据时，数据被写入缓冲器，并在最多 2 个时钟（32.768 kHz）后写入计数器。以 BCD 码设置 00 至 31 的十进制数到该寄存器。如果设置值超出该范围，在 1 个周期后寄存器的值将返回到正常值。
可以使用 8 位存储器操作指令来设置 DAY。
复位信号的产生会将该寄存器清除为 01H。

图 9-10. 日计数寄存器（DAY）的格式

地址：FF66H 复位后：01H R/W

符号	7	6	5	4	3	2	1	0
DAY	0	0	DAY20	DAY10	DAY8	DAY4	DAY2	DAY1

（10） 星期计数寄存器（WEEK）

WEEK 寄存器是 8 位寄存器，它的值是 0 至 6（十进制），用于表示星期的计数值。

日计数器溢出时，它将向上计数。

向该寄存器写入数据时，数据被写入缓冲器，并在最多 2 个时钟（32.768 kHz）后写入计数器。以 BCD 码设置 00 至 06 的十进制数到该寄存器。如果设置值超出该范围，在 1 个周期后寄存器的值将返回到正常值。

可以使用 8 位存储器操作指令来设置 WEEK。

复位信号的产生会将该寄存器清除为 00H。

图 9-11. 星期计数寄存器（WEEK）的格式

地址： FF65H 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
WEEK	0	0	0	0	0	WEEK4	WEEK2	WEEK1

<R> 注意事项 和月计数寄存器或日计数寄存器的对应关系，不在星期计数寄存器中自动存储。在复位释放后，星期计数寄存器设置如下。

Day	WEEK
星期天	00H
星期一	01H
星期二	02H
星期三	03H
星期四	04H
星期五	05H
星期六	06H

（11） 月计数寄存器（MONTH）

MONTH 寄存器是 8 位寄存器，它的值是 0 至 12（十进制），用于表示月的计数值。

日计数器溢出时，它将向上计数。

向该寄存器写入数据时，数据被写入缓冲器，并在最多 2 个时钟（32.768 kHz）后写入计数器。以 BCD 码设置 00 至 12 的十进制数到该寄存器。如果设置值超出该范围，在 1 个周期后寄存器的值将返回到正常值。

可以使用 8 位存储器操作指令来设置 MONTH。

复位信号的产生会将该寄存器清除为 01H。

图 9-12. 月计数寄存器（MONTH）的格式

地址：FF67H　复位后：01H　R/W

符号	7	6	5	4	3	2	1	0
MONTH	0	0	0	MONTH10	MONTH8	MONTH4	MONTH2	MONTH1

（12） 年计数寄存器（YEAR）

YEAR 寄存器是 8 位寄存器，它的值是 0 至 99（十进制），用于表示年的计数值。

月计数器溢出时，它将向上计数。

计数值为 00，04，08，…，92 和 96，表示是闰年。

向该寄存器写入数据时，数据被写入缓冲器，并在最多 2 个时钟（32.768 kHz）后写入计数器。以 BCD 码设置 00 至 99 的十进制数到该寄存器。如果设置值超出该范围，在 1 个周期后寄存器的值将返回到正常值。

可以使用 8 位存储器操作指令来设置 YEAR。

复位信号的产生会将该寄存器清除为 00H。

图 9-13. 年计数寄存器（YEAR）的格式

地址：FF68H　复位后：00H　R/W

符号	7	6	5	4	3	2	1	0
YEAR	YEAR80	YEAR40	YEAR20	YEAR10	YEAR8	YEAR4	YEAR2	YEAR1

<R>

（13） 钟表计数寄存器（SUBCUD）

当钟表快或慢的时候，通过改变从副计数寄存器（RSUBC）向秒计数寄存器溢出的计数值（参考值：7FFFH），该寄存器用于对钟表进行高精度修正。

可以使用 8 位存储器操作指令来设置 SUBCUD。

复位信号的产生会将该寄存器清除为 00H。

图 9-14. 钟表计数寄存器（SUBCUD）的格式

地址：FF82H　复位后：00H　R/W

符号	7	6	5	4	3	2	1	0
SUBCUD	DEV	F6	F5	F4	F3	F2	F1	F0

DEV	设置钟表修正时间
0	当秒数字为 00、20 或 40 时，修正钟表误差（每隔 20 秒）。
1	仅当秒数字为 00 时，修正钟表误差（每隔 60 秒）。

F6	设置钟表修正值
0	增加 {（F5，F4，F3，F2，F1，F0）– 1} × 2
1	减少 {（/F5，/F4，/F3，/F2，/F1，/F0）+ 1} × 2

当 (F6，F5，F4，F3，F2，F1，F0) = (*，0，0，0，0，0，*)时，钟表误差不做修正。 * 是 0 或 1。
/F5 至 /F0 是对应位的取反值 (当 111100 时，为 000011)
修正值的范围：　(当 F6 = 0 时)　2，4，6，8，…，120，122，124
　　　　　　　　(当 F6 = 1 时)　–2，–4，–6，–8，…，–120，–122，–124

通过钟表误差修正寄存器(SUBCUD)可以修正的值的范围如下所示。

	DEV = 0 (每20秒修正)	DEV = 1 (每60秒修正)
可修正的范围	–189.2 ppm 至 189.2 ppm	–63.1 ppm 至 63.1 ppm
最大量化误差	± 1.53 ppm	± 0.51 ppm
最小分辨率	± 3.05 ppm	± 1.02 ppm

备注　当修正范围是 –63.1 ppm 及更低时，或者 63.1 ppm 及更高时，设置 DEV 为 0。

（14） 闹钟分钟寄存器（ALARMWM）

该寄存器用于设置闹钟的分钟时间。

可以使用 8 位存储器操作指令来设置 ALARMWM。

复位信号的产生会将该寄存器清除为 00H。

注意事项　以 **BCD** 码设置 **00** 至 **59** 十进制值到该寄存器。如果设置的值超出范围，无法检测闹钟。

图 9-15. 闹钟分钟寄存器（ALARMWM）的格式

地址：FF86H　复位后：00H　R/W

符号	7	6	5	4	3	2	1	0
ALARMWM	0	WM40	WM20	WM10	WM8	WM4	WM2	WM1

（15） 闹钟小时寄存器（ALARMWH）

该寄存器用于设置闹钟的小时时间。

可以使用 8 位存储器操作指令来设置 ALARMWH。

复位信号的产生会将该寄存器清除为 12H。

<R>　但是，如果复位后 AMPM 位被置为 1，该寄存器的值是 00H。

注意事项　以 **BCD** 码设置 **00** 至 **23**、**01** 至 **12** 或 **21** 至 **32** 十进制值到该寄存器。如果设置的值超出范围，无法检测闹钟。

图 9-16. 闹钟小时寄存器（ALARMWH）的格式

地址：FF87H　复位后：12H　R/W

符号	7	6	5	4	3	2	1	0
ALARMWH	0	0	WH20	WH10	WH8	WH4	WH2	WH1

注意事项　当 **AMPM = 0**（如果选择 **12** 小时系统）时，**ALARMWH** 的第 **5** 位（**WH20**）表示 **AM**（**0**）**/PM**（**1**）。

（16） 闹钟星期寄存器（ALARMWW）

该寄存器用于设置闹钟的星期日期。

可以使用 8 位存储器操作指令来设置 ALARMWW。

复位信号的产生会将该寄存器清除为 00H。

图 9-17. 闹钟星期寄存器（ALARMWW）的格式

地址：FF88H　复位后：00H　R/W

符号	7	6	5	4	3	2	1	0
ALARMWW	0	WW6	WW5	WW4	WW3	WW2	WW1	WW0

以下是设置闹钟的示例。

闹钟时间	日							12 小时显示				24 小时显示			
	星期日 WW0	星期一 WW1	星期二 WW2	星期三 WW3	星期四 WW4	星期五 WW5	星期六 WW6	小时 10	小时 1	分钟 10	分钟 1	小时 10	小时 1	分钟 10	分钟 1
每天，0 : 00 a.m.	1	1	1	1	1	1	1	1	2	0	0	0	0	0	0
每天，1 : 30 a.m.	1	1	1	1	1	1	1	0	1	3	0	0	1	3	0
每天，11 : 59 a.m.	1	1	1	1	1	1	1	1	1	5	9	1	1	5	9
星期一到星期五，0 : 00 p.m.	0	1	1	1	1	1	0	3	2	0	0	1	2	0	0
星期天，1 : 30 p.m.	1	0	0	0	0	0	0	2	1	3	0	1	3	3	0
星期一、星期三、星期五，11 : 59 p.m.	0	1	0	1	0	1	0	3	1	5	9	2	3	5	9

9.4 实时计数器的操作

9.4.1 实时计数器的启动操作

图 **9-18.** 实时计数器的启动操作流程图



<R>　注　在 RTCE = 1 之后没有等待 INTRTC = 1 就转到 STOP 模式时，确认 **9.4.2 启动操作后转到 STOP 模式**中描述的过程。

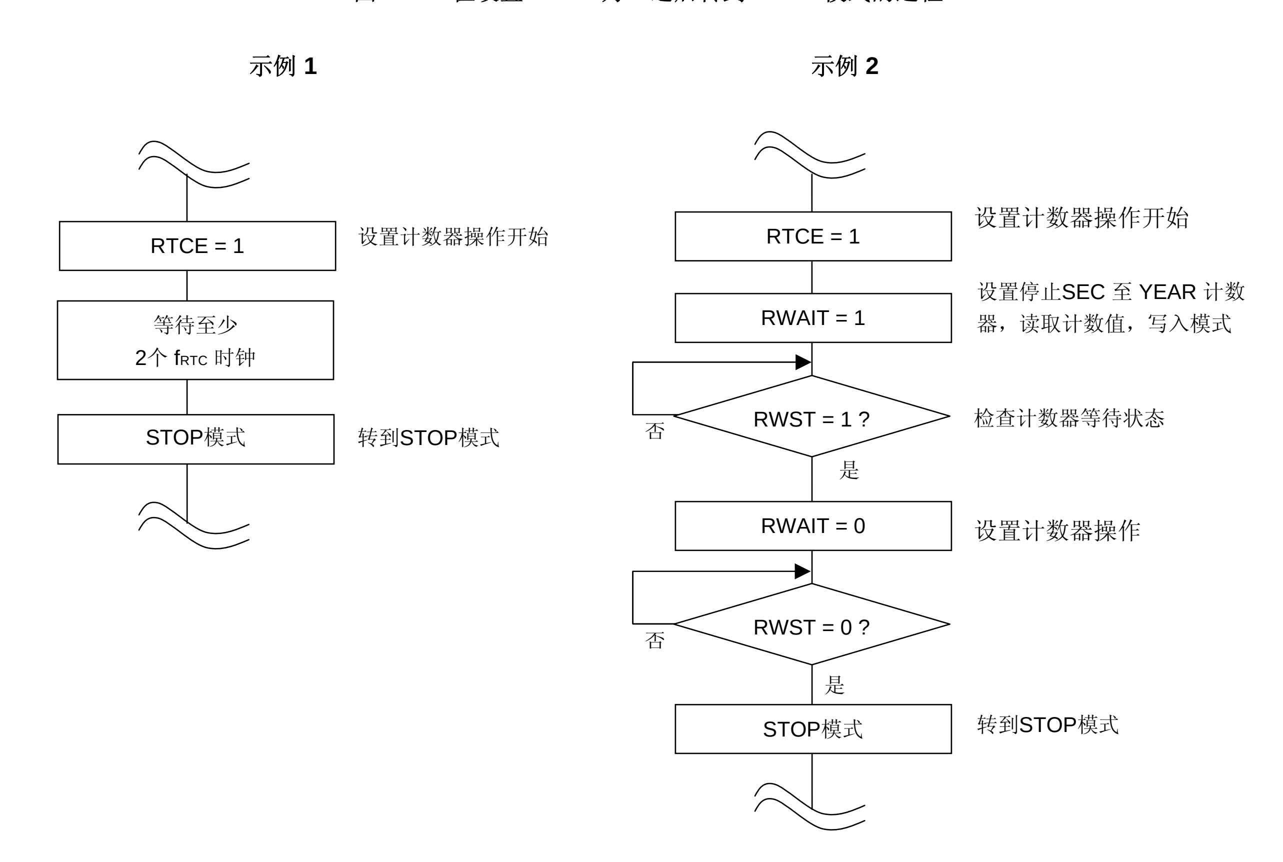
<R>

9.4.2 启动操作后转到STOP模式

在 RTCE = 1 之后立即转到 STOP 模式时，执行下列处理中的其中一个。
但是，在设置 RTCE 为 1 之后，在产生第一个 INTRTC 中断之后，转到 STOP 模式不需要这样的处理。

- 在设置 RTCE 为 1 之后，经过至少两个输入时钟(f_{RTC})后，转到 STOP 模式(参见图 **9-19**，示例 **1**)。
- 在设置 RTCE 为 1，然后设置 RWAIT 为 1 之后，轮询检查到 RWST 变为 1。此后，设置 RWAIT 为 0，并且通过轮询检查 RWST 再次变为 0 之后，转到 STOP 模式(参见图 **9-19**，示例 **2**)。

图 9-19. 在设置 RTCE 为 1 之后转到 STOP 模式的过程



9.4.3 读/写实时计数器

当向 RWAIT 设置 1 之后，读取或写入计数器。

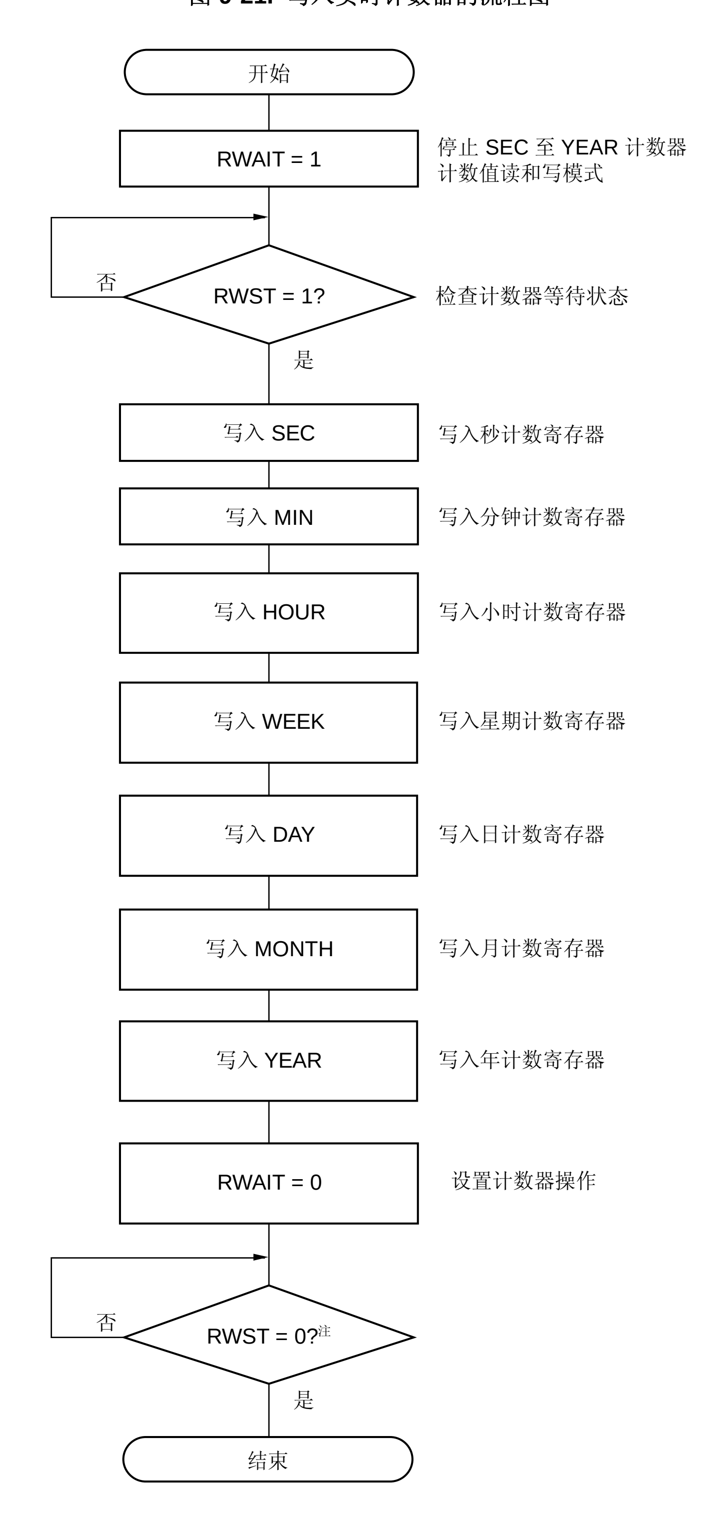
图 9-20. 读取实时计数器的流程图

注　　设置 STOP 模式前，必须确认 RWST = 0 。

注意事项　在 1 秒内完成一系列操作，将 RWAIT 置为 1 到清除 RWAIT 为 0 的操作。

备注　　SEC，MIN，HOUR，WEEK，DAY，MONTH 和 YEAR 可以按任意顺序读取。
不用设置所有的寄存器，仅某些寄存器可能被读取。

图 **9-21.** 写入实时计数器的流程图



注　　设置 STOP 模式前，必须确认 RWST = 0 。

注意事项　**在 1 秒内完成一系列操作，将 RWAIT 置为 1 到清除 RWAIT 为 0 的操作。**

备注　　SEC，MIN，HOUR，WEEK，DAY，MONTH 和 YEAR 可以按任意顺序读取。
不用设置所有的寄存器，仅某些寄存器可能被读取。

9.4.4 设置实时计数器的闹钟

首先设置 WALE 为 0 之后，设置闹钟时间。

图 9-22. 闹钟设置流程图

开始
WALE = 0　　闹钟匹配操作无效
WALIE = 1　　当闹钟匹配时产生中断
设置 ALARMWM　　设置闹钟分钟寄存器
设置 ALARMWH　　设置闹钟小时寄存器
设置 ALARMWW　　设置闹钟星期寄存器
WALE = 1　　闹钟匹配操作有效
INTRTC = 1?　否 / 是
WAFG = 1?　否 / 是
检测闹钟匹配
闹钟处理
固定周期中断服务

备注 **1.** ALARMWM，ALARMWH 和 ALARMWW 可以按任意顺序写入。

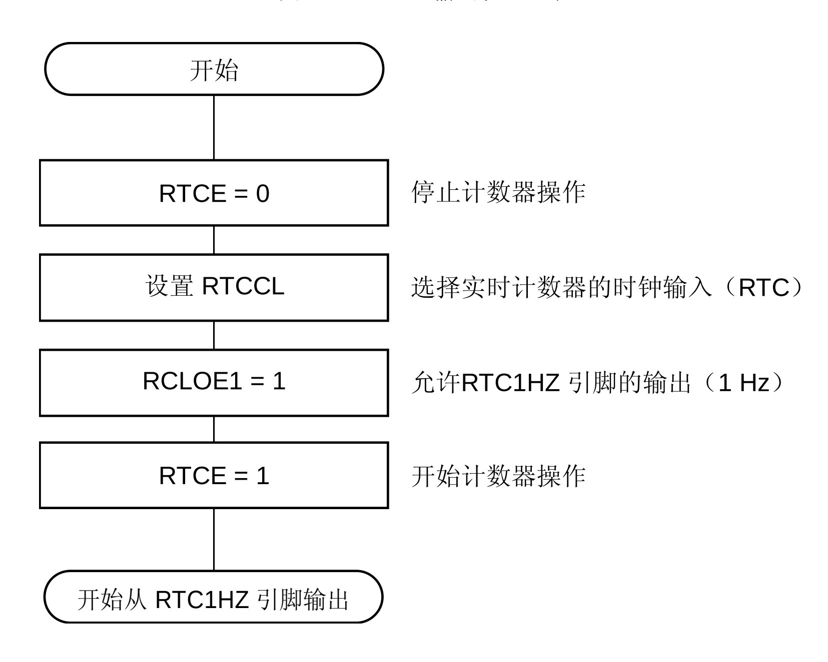
2. 固定周期中断和闹钟匹配中断使用相同的中断源（INTRTC）。当同时使用这两种类型中断时，在发生 INTRTC 时，检查固定周期中断状态标志（RIFG）和闹钟检测状态标志（WAFG），来判断是哪种中断发生

<R>

9.4.5 实时计数器的 1 Hz 输出

首先设置 RTCE 为 0 后，设置 1 Hz 输出。

图 9-23. 1 Hz 输出设置过程

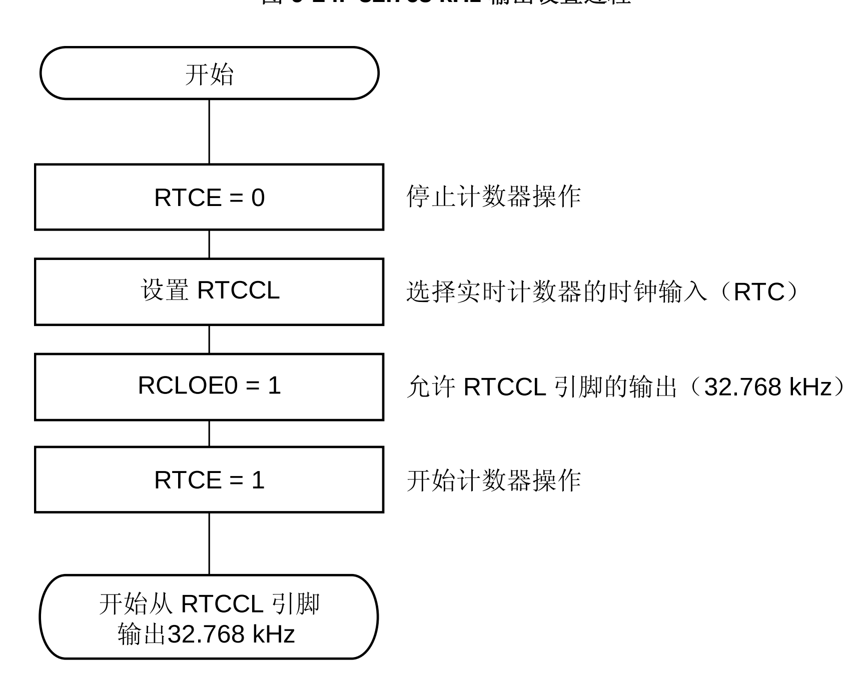


<R>

9.4.6 实时计数器的 32.768 kHz输出

首先设置 RTCE 为 0 后，设置 32.768 kHz 输出。

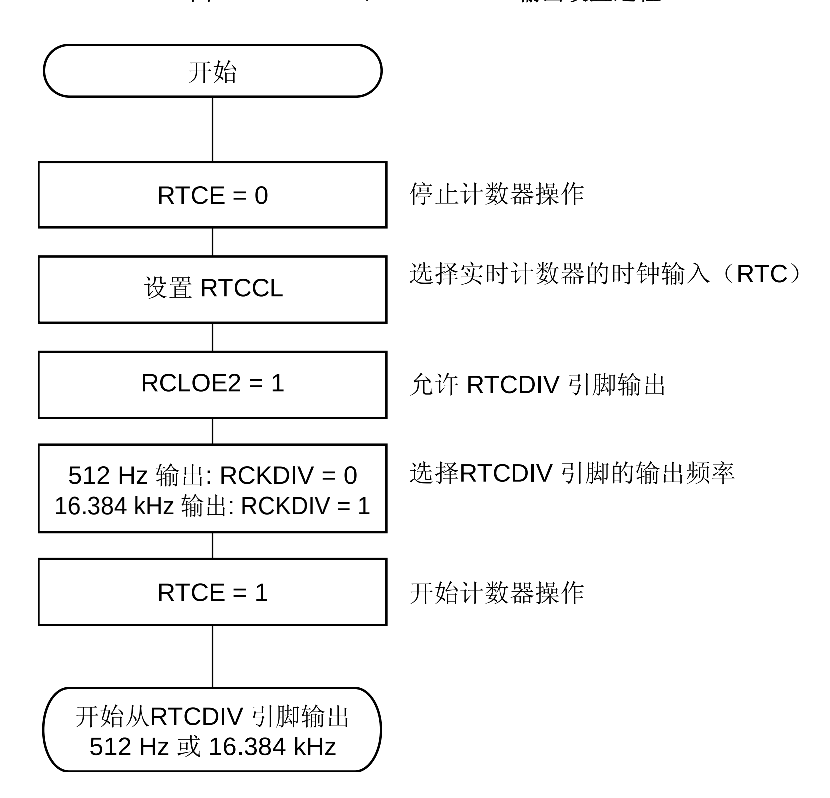
图 9-24. 32.768 kHz 输出设置过程



<R> **9.4.7 实时计数器的 512 Hz，16.384 kHz输出**

首先设置 RTCE 为 0 后，设置 512 Hz 或 16.384 kHz 输出。

图 9-25. 512 Hz，16.384 kHz 输出设置过程



<R>

9.4.8 实时计数器的钟表误差修正的示例

当钟表快或慢的时候，通过设置钟表误差修正寄存器的值，可以对钟表进行高精度修正。

计算修正值的示例

当修正副计数寄存器 (RSUBC)的计数值时，通过下面的表达式计算使用的修正值。

当修正范围是 –63.1 ppm 及更低时，或者 63.1 ppm 及更高时，设置 DEV 为 0。

(当 DEV = 0 时)

修正值[注] = 1 分钟内修正计数的数量 ÷ 3 = (振荡频率 ÷ 目标频率 – 1) × 32768 × 60 ÷ 3

(当 DEV = 1 时)

修正值[注] = 1 分钟内修正计数的数量 = (振荡频率 ÷ 目标频率 – 1) × 32768 × 60

注　修正值是钟表误差修正值，通过钟表误差修正寄存器(SUBCUD)的第 6 位至第 0 位计算得到。

(当 F6 = 0) 修正值 = {(F5，F4，F3，F2，F1，F0) – 1} × 2

(当 F6 = 1) 修正值 = – {(/F5，/F4，/F3，/F2，/F1，/F0) + 1} × 2

当 (F6，F5，F4，F3，F2，F1，F0) 是 (*，0，0，0，0，0，*)时，不执行钟表误差修正。

"*" 是 0 或 1。

/F5 至 /F0 是逐位取反的值 (当 111100 时，为 000011)

备注

1. 修正值是 2，4，6，8，… 120，122，124 或 –2，–4，–6，–8，… –120，–122，–124.

2. 振荡频率是实时计数器 (RTC)的输入时钟(f_{RTC})值。
当钟表误差修正寄存器被设置为初始值(00H)时，可以从 RTCCL 引脚的 32 kHz 输出频率，或者 RTC1HZ 引脚的输出频率 × 32768 来计算该值。

3. 目标频率是使用钟表误差修正寄存器执行修正后的频率结果。

修正示例 <1>

从 32772.3 Hz 修正到 32768 Hz (32772.3 Hz – 131.2 ppm)的示例

[测量振荡频率]

当钟表误差修正寄存器被设置为初始值(00H)时，各个产品的振荡频率[注] 可以通过从 RTCCL 引脚输出的 32 kHz 或者从 RTC1HZ 引脚输出的 1Hz 来测量。

注　关于从 RTC1HZ 引脚输出约 1Hz 的设置过程，参见 **9.4.5　实时计数器的 1 Hz 输出**，关于从 RTCCL 引脚输出约 32kHz 的设置过程，参见 **9.4.6 实时计数器的 32.768 kHz 输出**。

[计算修正值]

(当从 RTCCL 引脚输出的频率是 32772.3 Hz)

如果目标频率假设是 32768 Hz (32772.3 Hz – 131.2 ppm)，对 –131.2 ppm 的修正范围是 –63.1 ppm 或更少，于是假设 DEV 为 0。

当假设 DEV 为 0 时，计算修正值的表达式如下。

修正值 = 1 分钟内修正计数的数量 ÷ 3
= (振荡频率 ÷ 目标频率 – 1) × 32768 × 60 ÷ 3
= (32772.3 ÷ 32768 – 1) × 32768 × 60 ÷ 3
= 86

[计算要设置到 (F6 至 F0)的值]

(当修正值是 86 时)

如果修正值是 0 或更大的值(当延迟减慢时)，假设 F6 为 0。

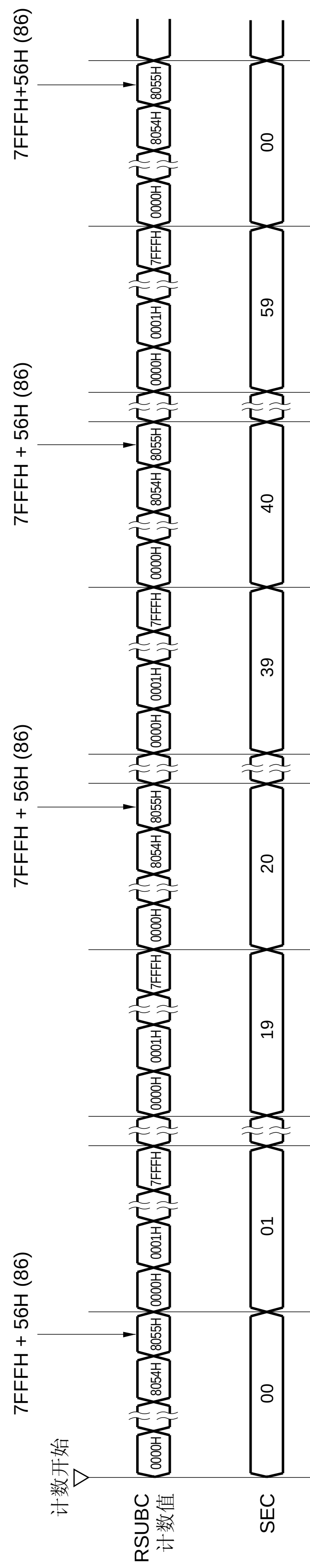
由修正值计算 (F5，F4，F3，F2，F1，F0)。

{ (F5，F4，F3，F2，F1，F0) – 1} × 2 = 86
(F5，F4，F3，F2，F1，F0) = 44
(F5，F4，F3，F2，F1，F0) = (1，0，1，1，0，0)

于是，当从 32772.3 Hz 修正到 32768 Hz (32772.3 Hz – 131.2 ppm)时，设置修正寄存器，比如 DEV 是 0 且修正值是 86 (SUBCUD 的第 6 位至第 0 位：0101100) 会导致 32768 Hz (0 ppm)。

图 9-26 展示了当 (DEV，F6，F5，F4，F3，F2，F1，F0) 是 (0，0，1，0，1，1，0，0)时的操作。

图 9-26. 当(DEV，F6，F5，F4，F3，F2，F1，F0) = (0，0，1，0，1，1，0，0)时的操作



修正示例 <2>

从 32767.4 Hz 修正到 32768 Hz (32767.4 Hz + 18.3 ppm)的示例

[测量振荡频率]

当钟表误差修正寄存器被设置为初始值(00H)时，各个产品的振荡频率[注] 可以通过从 RTCCL 引脚输出的 32 kHz 或者从 RTC1HZ 引脚输出的 1Hz 来测量。

注　关于从 RTC1HZ 引脚输出约 1Hz 的设置过程，参见 **9.4.5　实时计数器的 1 Hz 输出**，关于从 RTCCL 引脚输出约 32kHz 的设置过程，参见 **9.4.6 实时计数器的 32.768 kHz 输出**。

[计算修正值]

(当从 RTCCL 引脚输出的频率是 0.9999817 Hz)

振荡频率 = 32768 × 0.9999817 ≈ 32767.4 Hz

假设目标频率是 32768 Hz (32767.4 Hz + 18.3 ppm)，且 DEV 为 1。

当假设 DEV 为 1 时，计算修正值的表达式如下。

修正值 = 1 分钟内修正计数的数量

= (振荡频率 ÷ 目标频率 – 1) × 32768 × 60

= (32767.4 ÷ 32768 – 1) × 32768 × 60

= –36

[计算要设置到 (F6 至 F0)的值]

(当修正值是–36 时)

如果修正值是 0 或更小的值(当加速时)，假设 F6 为 1。

由修正值计算 (F5，F4，F3，F2，F1，F0)。

– {(/F5，/F4，/F3，/F2，/F1，/F0) + 1} × 2　= –36

(/F5，/F4，/F3，/F2，/F1，/F0)　= 17

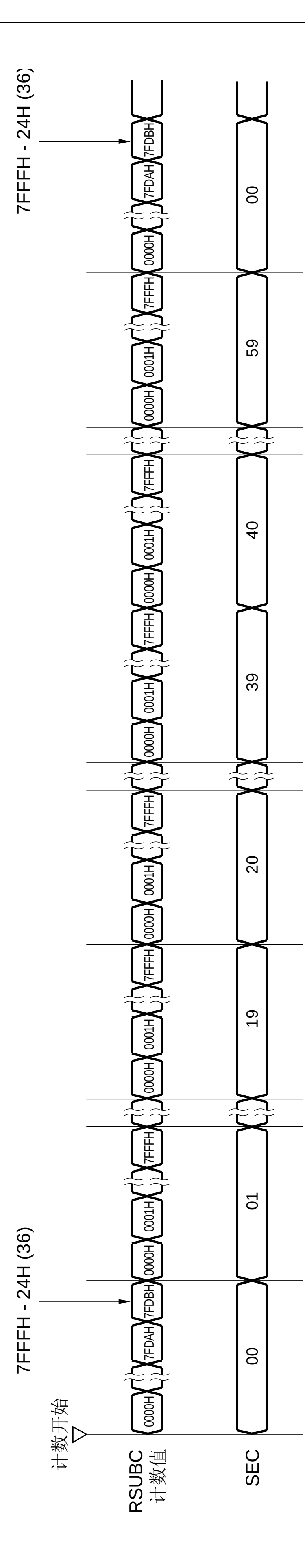
(/F5，/F4，/F3，/F2，/F1，/F0)　= (0，1，0，0，0，1)

(F5，F4，F3，F2，F1，F0)　= (1，0，1，1，1，0)

于是，当从 32767.4 Hz 修正到 32768 Hz (32767.4 Hz + 18.3 ppm)时，设置修正寄存器，比如 DEV 是 1 且修正值是–36 (SUBCUD 的第 6 位至第 0 位：1101110) 会导致 32768 Hz (0 ppm)。

图 9-27 展示了当 (DEV，F6，F5，F4，F3，F2，F1，F0) 是 (1，1，1，0，1，1，1，0)时的操作。

图 9-27. 当 (DEV，F6，F5，F4，F3，F2，F1，F0) = (1，1，1，0，1，1，1，0)时的操作



第十章　看门狗定时器

10.1 看门狗定时器的功能

看门狗定时器运行于内部低速振荡时钟。.

看门狗定时器用于检测期望之外的程序循环。如果检测到一个程序循环，产生一个内部复位信号。

出现以下情况时认为检测到程序循环。

- 如果看门狗定时器计数器溢出
- 如果对看门狗定时器使能寄存器（WDTE）执行 1 位操作指令。
- 如果将“ACH”以外的数据写入 WDTE
- 如果在窗口关闭期间将数据写入 WDTE
- 如果从 IMS 和 IXS 寄存器设置的区域之外获取指令（在 CPU 挂起时检测到无效检查）
- 如果因为执行 1 条读/写指令导致 CPU 访问 IMS 和 IXS 寄存器设置的区域之外（FB00H 至 FFFFH 除外）（在 CPU 程序循环期间检测到非法访问）

当由看门狗定时器引发复位时，复位控制标志寄存器（RESF）的第 4 位（WDTRF）置 1。RESF 的详细信息，参见 第二十三章 复位功能。

10.2 看门狗定时器的配置

看门狗定时器包含以下硬件。

表 10-1. 看门狗定时器的配置

项目	配置
控制寄存器	看门狗定时器使能寄存器 （WDTE）

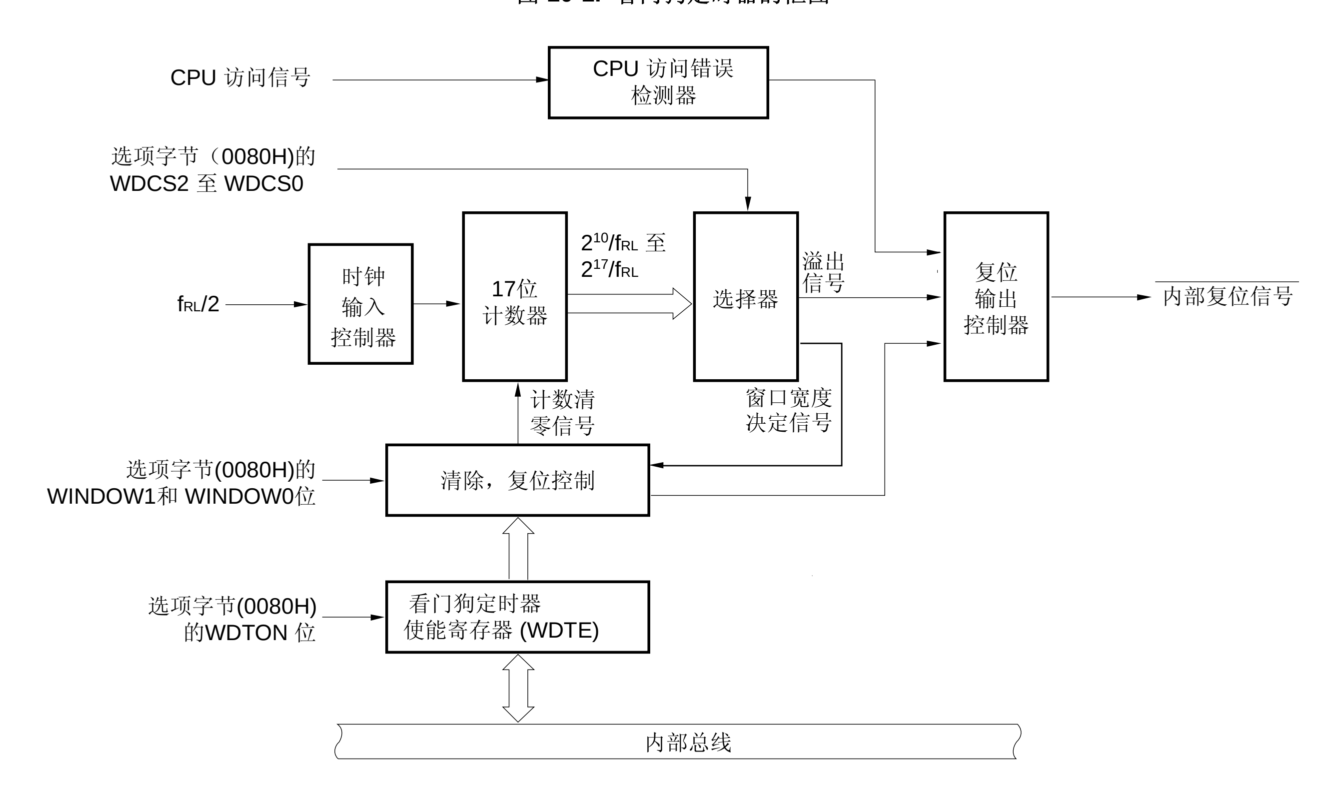
通过选项字节设置如何控制计数器操作、溢出时间和窗口的打开时期。

表 10-2. 选项字节和看门狗定时器的设置

看门狗定时器的设置	选项字节（0080H）
窗口打开时期	第6位 和 第5位（WINDOW1，WINDOW0）
控制看门狗定时器的计数器操作	第4位（WDTON）
看门狗定时器的溢出时间	第3位 至 第1位（WDCS2 至 WDCS0）

备注　　需要了解选项字节的内容，参见 **第二十六章 选项字节**。

图 10-1. 看门狗定时器的框图



10.3 控制看门狗定时器的寄存器

看门狗定时器由看门狗定时器使能寄存器（WDTE）控制。

（1）看门狗定时器使能寄存器（WDTE）

向 WDTE 写入 ACH，可以清除看门狗定时器计数器，并再次开始计数。

可以通过 8 位存储器操作指令来设置该寄存器。

复位信号的产生会将该寄存器设置为 9AH 或 1AH [注]。

图 **10-2.** 看门狗定时器使能寄存器 （**WDTE**） 的格式

地址: FF99H　复位后: 9AH/1AH[注]　R/W

符号	7	6	5	4	3	2	1	0
WDTE								

注　WDTE 的复位值取决于选项字节（0080H）中 WDTON 位的设置值。 要操作看门狗定时器，设置 WDTON 为 1。

WDTON设置值	WDTE复位值
0 （禁止看门狗定时器计数的操作）	1AH
1 （使能看门狗定时器计数的操作）	9AH

注意事项
1. **如果写入 WDTE 的值不是 ACH，产生一个内部复位信号。如果看门狗定时器的时钟源被停止，则当看门狗定时器的时钟源恢复操作时，产生内部复位信号。**
2. **如果使用 1 位存储器操作指令对 WDTE 进行操作，产生一个内部复位信号。如果看门狗定时器的时钟源被停止，则当看门狗定时器的时钟源恢复操作时，产生内部复位信号。**
3. **从 WDTE 读取的值为 9AH/1AH（与写入值（ACH）不同）。**

10.4 看门狗定时器的操作

10.4.1 看门狗定时器的控制操作

1. 当使用看门狗定时器时，由选项字节（0080H）指定其操作。

 - 通过设置选项字节（0080H）的第 4 位（WDTON）为 1，使能看门狗定时器的计数操作（复位释放后计数器开始计数）（详情参见 第二十六章）。

WDTON	看门狗定时器计数器的操作控制/非法访问检测
0	禁止计数器操作 （复位后停止计数），禁止非法访问的检测操作
1	使能计数器操作 （复位后开始计数），使能非法访问的检测操作

 - 通过选项字节（0080H）的第 3 位至第 1 位（WDCS2 至 WDCS0）设置溢出时间（详情参见 **10.4.2** 和 第二十六章）。
 - 通过选项字节（0080H）的第 6 位和第 5 位（WINDOW1 和 WINDOW0）设置窗口打开时期（详情参见 **10.4.3** 和 第二十六章）。

2. 复位释放后，看门狗定时器开始计数。
3. 在看门狗定时器开始计数后，且在达到选项字节设置的溢出时间之前，将“ACH”写入 WDTE，看门狗定时器被清除并再次开始计数。
4. 此后，在复位释放后的窗口打开时期，第二次及更多次写入 WDTE。如果在窗口关闭期间写 WDTE，则会产生内部复位信号。
5. 如果已经经过溢出时间，还没有将“ACH”写入 WDTE，则会产生内部复位信号。
   在以下情况时，会产生内部复位信号。

 - 如果对看门狗定时器使能寄存器（WDTE）执行 1 位操作指令时。
 - 如果将“ACH”以外的数据写入 WDTE。
 - 如果从 IMS 和 IXS 寄存器设置的区域之外获取指令（在 CPU 挂起时检测到无效检查）
 - 如果因为执行 1 条读/写指令导致 CPU 访问 IMS 和 IXS 寄存器设置的区域之外（FB00H 至 FFFFH 除外）（在 CPU 程序循环期间检测到非法访问）

注意事项 **1. 复位释放后对 WDTE 进行的第一次写操作，清除看门狗定时器，如果这一操作是在到达溢出时间之前进行的，而不论写入操作的时序，看门狗定时器会再次开始计数。**

2. 如果通过将“ACH”写入 WDTE 来清除看门狗定时器，则实际的溢出时间可能与由选项字节设置的溢出时间不同，最大差距可达 2/f_{RL} 秒。

3. 在计数值马上溢出（FFFFH）之前，都可以清除看门狗定时器。

注意事项　**4.** 在 **HALT** 和 **STOP** 模式下看门狗定时器的操作会根据选项字节第 **0** 位（**LSROSC**）的设置值而不同，如下所示。

	LSROSC = 0 （内部低速振荡器可由软件停止）	LSROSC = 1 （不能停止内部低速振荡器）
HALT 模式	看门狗定时器操作停止	看门狗定时器操作继续
STOP 模式		

如果 **LSROSC = 0**，则在 **HALT** 或 **STOP** 模式释放后，看门狗定时器恢复计数。此时，计数器不被清除为 **0**，而是从计数器停止时的值开始计数。

当 **LSROSC = 0** 时，如果通过设置 **LSRSTOP**（内部振荡模式寄存器（**RCM**）的第 **1** 位**=1**）停止内部低速振荡器的振荡，则看门狗定时器停止操作。此时，计数器不被清除为 **0**。

5. 在 **flash** 存储器闪存自编程和 **EEPROM**™ 模拟期间，看门狗定时器继续操作。在处理期间，中断响应时间被延迟。设置溢出时间和窗口大小时，应考虑该延迟。

10.4.2 设置看门狗定时器的溢出时间

通过选项字节（0080H）的第 3 位至第 1 位（WDCS2 至 WDCS0），设置看门狗定时器的溢出时间。

如果发生溢出，则产生内部复位信号。在到达溢出时间之前的窗口打开期间，将“ACH”写入 WDTE，可以清除当前计数值，且看门狗定时器再次开始计数。

溢出时间设置如下。

表 10-3. 看门狗定时器溢出时间的设置

WDCS2	WDCS1	WDCS0	看门狗定时器的溢出时间
0	0	0	$2^{10}/f_{RL}$ （3.88 ms）
0	0	1	$2^{11}/f_{RL}$ （7.76 ms）
0	1	0	$2^{12}/f_{RL}$ （15.52 ms）
0	1	1	$2^{13}/f_{RL}$ （31.03 ms）
1	0	0	$2^{14}/f_{RL}$ （62.06 ms）
1	0	1	$2^{15}/f_{RL}$ （124.12 ms）
1	1	0	$2^{16}/f_{RL}$ （248.24 ms）
1	1	1	$2^{17}/f_{RL}$ （496.48 ms）

注意事项　**1.** 禁止设置 **WDCS2 = WDCS1 = WDCS0 = 0** 且 **WINDOW1 = WINDOW0 = 0**。

2. 在 **flash** 存储器闪存自编程和 **EEPROM**™ 模拟期间，看门狗定时器继续操作。在处理期间，中断响应时间被延迟。设置溢出时间和窗口大小时，应考虑该延迟。

备注　**1.** f_{RL}：　内部低速振荡时钟频率

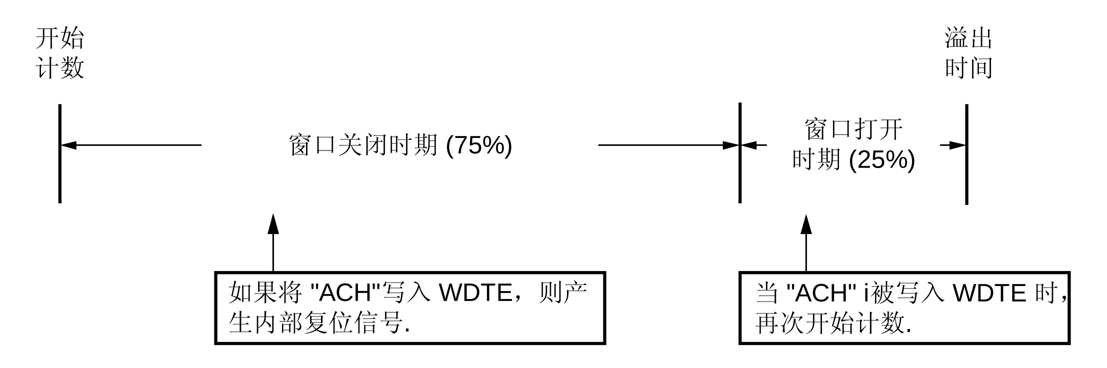
2. （ ）：　f_{RL} = 264 kHz （最大值）

10.4.3 设置看门狗定时器的窗口打开时期

通过选项字节（0080H）的第 6 位和第 5 位（WINDOW1，WINDOW0），设置看门狗定时器的窗口打开时期。窗口的概述如下。

- 如果在窗口打开期间将“ACH”写入 WDTE，则看门狗定时器被清除并再次开始计数。
- 即使在窗口关闭期间将“ACH”写入 WDTE，也会检测到异常，并产生内部复位信号。

举例： 如果窗口打开时期为 25%



注意事项　复位释放后对 WDTE 进行的第一次写操作，清除看门狗定时器，如果这一操作是在到达溢出时间之前进行的，而不论写入操作的时序，看门狗定时器会再次开始计数。

窗口打开时期的设置如下。

表 10-4. 设置看门狗定时器的窗口打开时期

WINDOW1	WINDOW0	看门狗定时器的窗口打开时期
0	0	25%
0	1	50%
1	0	75%
1	1	100%

注意事项 **1.** 禁止设置 **WDCS2 = WDCS1 = WDCS0 = 0** 且 **WINDOW1 = WINDOW0 = 0**。

2. 在 **flash** 存储器闪存自编程和 **EEPROM™** 模拟期间，看门狗定时器继续操作。在处理期间，中断响应时间被延迟。设置溢出时间和窗口大小时，应考虑该延迟。

<R>

备注　如果将溢出时间设置为 $2^{10}/f_{RL}$，则窗口关闭时间和打开时间如下所示。

（$2.6\ V \leq V_{DD} \leq 5.5\ V$）

	窗口打开时期的设置			
	25%	50%	75%	100%
窗口关闭时间	0 至 3.56 ms	0 至 2.37 ms	0 至 0.119 ms	无
窗口打开时间	3.56 至 3.88 ms	2.37 至 3.88 ms	0.119 至 3.88 ms	0 至 3.88 ms

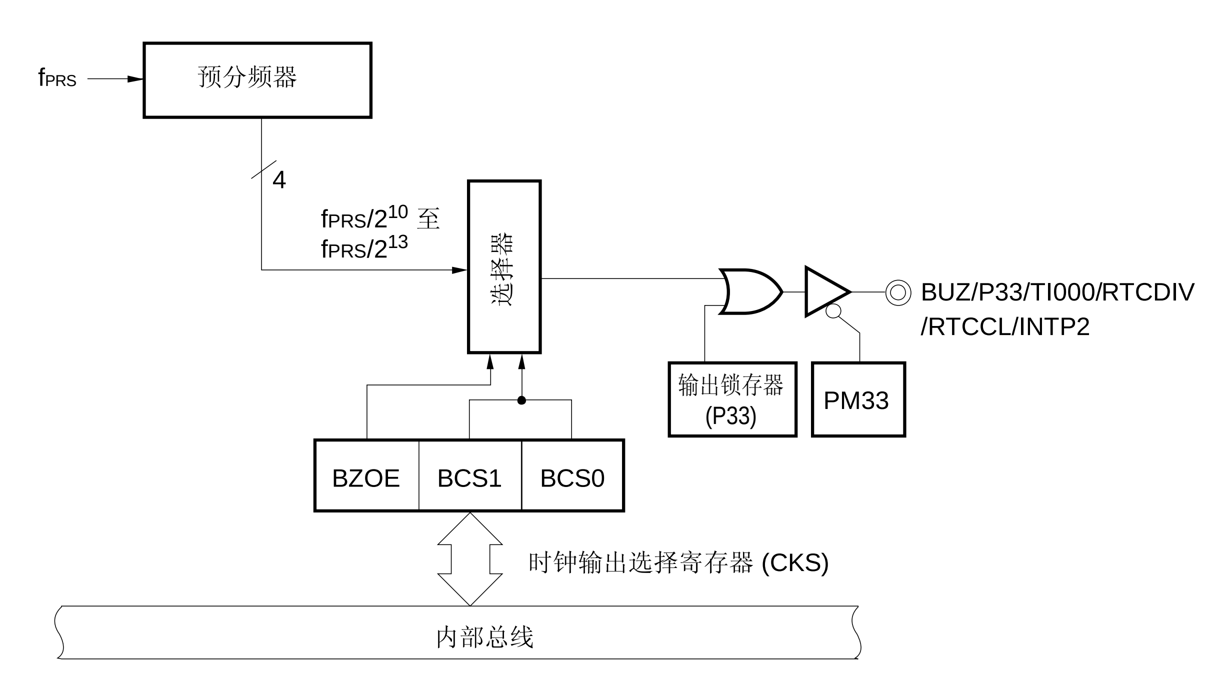
<当窗口打开时期为 25%>

- 溢出时间：
  $2^{10}/f_{RL}$（最大值） = 2^{10}/264 kHz（最大值） = 3.88 ms
- 窗口关闭时间：
  0 至 $2^{10}/f_{RL}$ （最小值） × （1 – 0.25） = 0 至 2^{10}/216 kHz （最小值） × 0.75 = 0 至 3.56 ms
- 窗口打开时间：
  $2^{10}/f_{RL}$ （最小值） × （1 – 0.25） 至 $2^{10}/f_{RL}$ （最大值） = 2^{10}/216 kHz （最小值） × 0.75 至 2^{10}/264 kHz （最大值）
  = 3.56 至 3.88 ms

11.1 蜂鸣器输出控制器的功能

蜂鸣器输出用于由 CKS 选择的蜂鸣器频率的方波输出。
图 11-1 展示了蜂鸣器输出控制器的框图。

图 11-1. 蜂鸣器输出控制器的框图



11.2 蜂鸣器输出控制器的配置

蜂鸣器输出控制器包括以下硬件。

表 11-1. 蜂鸣器输出控制器的配置

项目	配置
控制寄存器	时钟输出选择寄存器（CKS） 端口模式寄存器 3（PM3） 端口寄存器 3（P3）

11.3 控制蜂鸣器输出控制器的寄存器

以下两个寄存器用于控制蜂鸣器输出控制器。

- 时钟输出选择寄存器 （CKS）
- 端口模式寄存器 3（PM3）

（1） 时钟输出选择寄存器（CKS）

该寄存器用于设置蜂鸣器频率输出（BUZ）的输出使能/禁止，并设置输出时钟。

可以通过 1 位或 8 位存储器操作指令来设置 CKS。

复位信号的产生会将 CKS 清除为 00H。

图 11-2. 时钟输出选择寄存器的格式（CKS）

地址： FF40H　复位后：00H　R/W

符号	<7>	6	5	<4>	3	2	1	0
CKS	BZOE	BCS1	BCS0	0	0	0	0	0

BZOE	BUZ 输出使能/禁止规范
0	停止时钟分频电路的操作。BUZ 固定为低电平。
1	使能时钟分频电路的操作。BUZ 输出使能。

BCS1	BCS0	BUZ 输出时钟选择		
			f_{PRS} = 5 MHz	f_{PRS} = 10 MHz
0	0	$f_{PRS}/2^{10}$	4.88 kHz	9.77 kHz
0	1	$f_{PRS}/2^{11}$	2.44 kHz	4.88 kHz
1	0	$f_{PRS}/2^{12}$	1.22 kHz	2.44 kHz
1	1	$f_{PRS}/2^{13}$	0.61 kHz	1.22 kHz

注意事项　**当蜂鸣器输出操作被停止时（BZOE = 0），设置 BCS1 和 BCS0。**

备注　f_{PRS}：外设硬件时钟频率

（2） 端口模式寄存器 3 （PM3）

该寄存器用于按位设置端口 3 的输入/输出模式。

当使用 P33/TI000/RTCDIV/RTCCL/BUZ/INTP2 引脚作为蜂鸣器输出时，清除 PM33 并将 P33 的输出锁存清除为 0。

可以通过 1 位或 8 位存储器操作指令来设置 PM3 。

复位信号的产生会将 PM3 设置为 FFH。

图 11-3. 端口模式寄存器 3（PM3）的格式

地址： FF23H　复位后： FFH　R/W

符号	7	6	5	4	3	2	1	0
PM3	1	1	1	PM34	PM33	PM32	PM31	1

PM3n	P3n 引脚 I/O 模式选择 （n = 1 至 4）
0	输出模式 （输出缓冲器打开）
1	输入模式 （输出缓冲器关闭）

11.4 蜂鸣器输出控制器的操作

蜂鸣器频率输出按照如下步骤进行。

<1> 通过时钟输出选择寄存器（CKS）的第 5 和 6 位 （BCS0，BCS1），选择蜂鸣器输出频率（蜂鸣器输出是禁止状态）。

<2> 设置 CKS 的第 7 位（BZOE）为 1，可以使能蜂鸣器输出。

第十二章 10 位逐次逼近型A/D转换器
（仅限μPD78F045x和 78F046x）

12.1 10 位逐次逼近型A/D转换器的功能

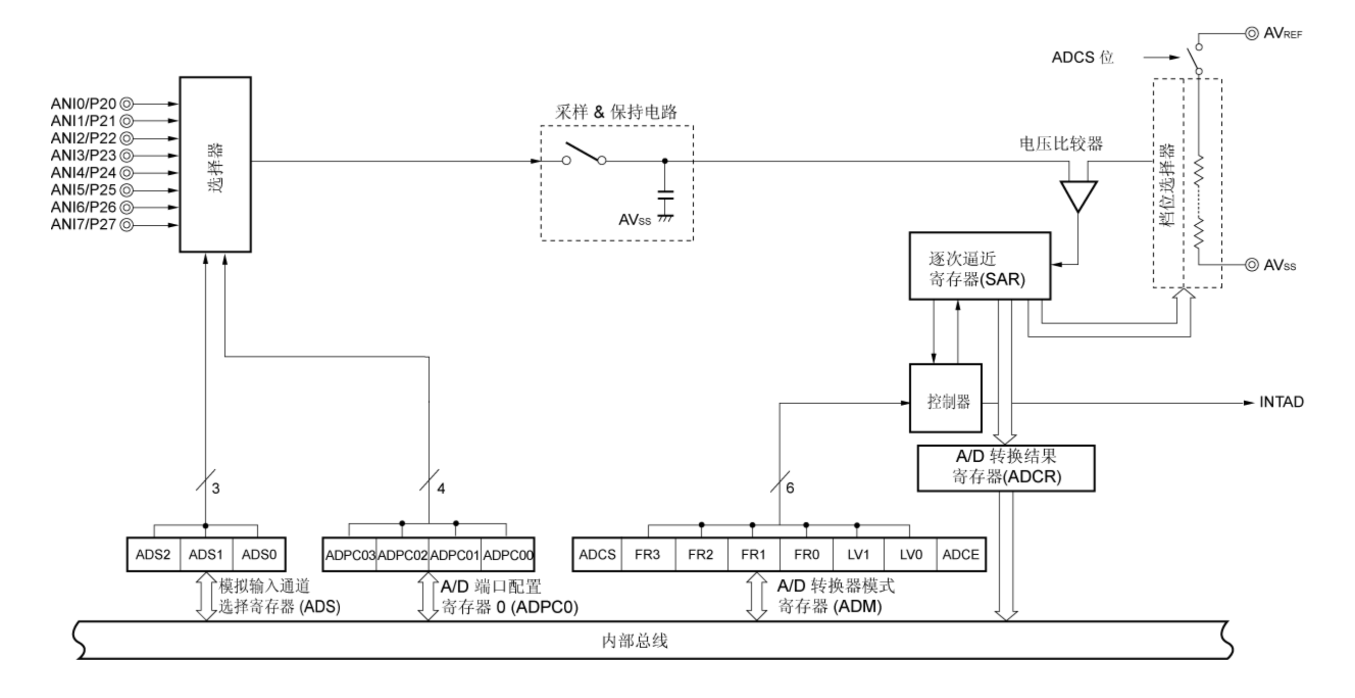
10 位逐次逼近型 A/D 转换器用于将模拟输入信号转换为数字信号，最多包括 8 个通道（ANI0 至 ANI7），具有 10 位分辨率。

A/D 转换器具有以下功能。

- **10 位分辨率 A/D 转换**

  从模拟输入通道 ANI0 至 ANI7 中选择其中一个，对其重复执行 10 位分辨率 A/D 转换。每次 A/D 转换结束，都一个中断请求（INTAD）。

图 12-1. 10 位逐次逼近型 A/D 转换器的框图



12.2 10 位逐次逼近型A/D转换器的配置

10 位逐次逼近型 A/D 转换器包括以下硬件。

（1） ANI0 至 ANI7 引脚

这些是 10 位逐次逼近型 A/D 转换器的 8 通道模拟输入引脚。从这些引脚输入的模拟信号被转变为数字信号。除了那个被选为模拟输入的引脚外，其他引脚均可用作 I/O 端口引脚或是 segment 输出引脚（仅限μPD78F045x）。

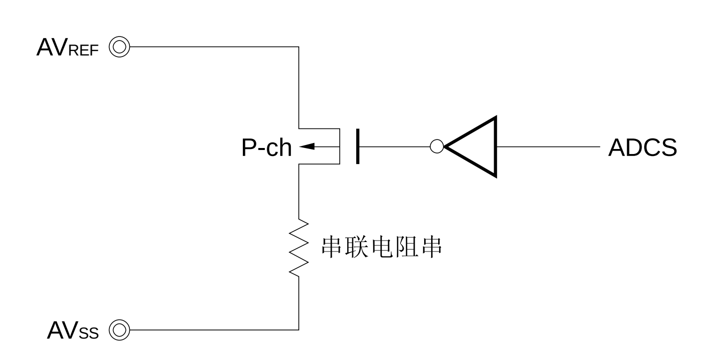
（2） 采样&保持电路

在 A/D 转换器启动时，采样&保持电路用于对选择器选择的模拟输入引脚的输入电压进行采样，并在 A/D 转换期间保持采样到的电压值。

（3） 串联电阻串

串联电阻串连接在 AV_{REF} 和 AV_{SS} 之间，并产生一个电压，用于和采样到的电压值进行比较。

图 12-2. 串联电阻串的电路配置



（4） 电压比较器

电压比较器用于将采样到的电压值与串联电阻串的输出电压进行比较。

（5） 逐次逼近寄存器 （SAR）

该寄存器用于转换电压比较器的比较结果，从最高有效位（MSB）开始。

当电压被转换为数字值直到最低有效位时（A/D 转换结束），SAR 寄存器的内容传送到 A/D 转换结果寄存器（ADCR）。

（6） 10 位 A/D 转换结果寄存器 （ADCR）

每当 A/D 转换结束时，A/D 转换结果从逐次逼近寄存器装载到该寄存器中。转换结果被保存在该寄存器的高 10 位中（低 6 位固定为 0）。

（7） 8 位 A/D 转换结果寄存器（ADCRH）

每当 A/D 转换结束时，A/D 转换结果从逐次逼近寄存器装载到该寄存器中。ACRH 保存转换结果的高 8 位。

注意事项 从 ADCR 和 ADCRH 读取数据时，产生一个等待周期。当 CPU 运行于副系统时钟，且外设硬件时钟处于停止状态时，不要从 ADCR 和 ADCRH 读取数据。详情参见第三十三章 等待注意事项。

（8） 控制器

该电路用于控制模拟输入信号转换为数字信号的转换时间，也可以控制转换操作的启动和停止。当 A/D 转换结束时，该控制器产生中断 INTAD。

（9） AV_{REF} 引脚

该引脚为 A/D 转换器输入一个模拟供电/参考电压。当端口 2 被用作数字端口时或用作 segment 输出时，应该设置该引脚与 V_{DD} 的电位相同。

基于 AV_{REF} 和 AV_{SS} 的提供的电压，输入到 ANI0 至 ANI7 的信号被转换为数字信号。

（10） AV_{SS} 引脚

这是 A/D 转换器的地电位引脚。即便没有使用 A/D 转换器，该引脚的电压始终应该和 V_{SS} 引脚的电位相同。

（11） A/D 转换器模式寄存器（ADM）

该寄存器用于设置转换模拟输入信号所需的转换时间，并设置转换操作的启动或停止。

（12） A/D 端口配置寄存器 0 （ADPC0）

该寄存器用于将 ANI0/P20 至 ANI7/P27 引脚切换为模拟输入（16 位 ΔΣ型 A/D 转换器的模拟输入或 10 位逐次逼近型 A/D 转换器的模拟输入）或端口的数字 I/O。

（13） 模拟输入通道选择寄存器（ADS）

该寄存器用于指定输入模拟电压的端口，该模拟信号将被转换为数字信号。

（14） 端口模式寄存器 2（PM2）

该寄存器用于将 ANI0/P20 至 ANI7/P27 引脚切换为输入或输出。

（15） 端口功能寄存器 2（PF2）（仅限μPD78F045x）

该寄存器用于切换 ANI0/P20 至 ANI7/P27 引脚为端口 I/O、A/D 转换器的模拟输入或 segment 输出。

12.3 10 位逐次逼近型A/D转换器使用的寄存器

A/D 转换器使用以下七个寄存器。

- A/D 转换器模式寄存器（ADM）
- A/D 端口配置寄存器 0（ADPC0）
- 模拟输入通道选择寄存器（ADS）
- 端口功能寄存器 2 （PF2）（仅限　PD78F045x）
- 端口模式寄存器 2（PM2）
- 10 位 A/D 转换结果寄存器（ADCR）
- 8 位 A/D 转换结果寄存器（ADCRH）

（1）　A/D 转换器模式寄存器（ADM）

该寄存器用于设置模拟输入进行 A/D 转换所需的转换时间，并设置转换操作的启动或停止。

可以通过 1 位或 8 位存储器操作指令来设置 ADM。

复位信号产生会将该寄存器清除为 00H。

图 12-3. A/D 转换器模式寄存器（ADM）的格式

地址: FF8DH　复位后: 00H　R/W

符号	<7>	6	5	4	3	2	1	<0>
ADM	ADCS	FR3[注 1]	FR2[注 1]	FR1[注 1]	FR0[注1]	LV1[注 1]	LV0[注 1]	ADCE

ADCS	A/D 转换操作控制
0	停止转换操作
1	允许转换操作

ADCE	比较器操作控制[注2]
0	停止比较器的操作
1	允许比较器的操作

注　**1.** FR3 至 FR0、LV1、LV0 和 A/D 转换的详细信息，参见**表 12-2 A/D 转换时间的选择**。

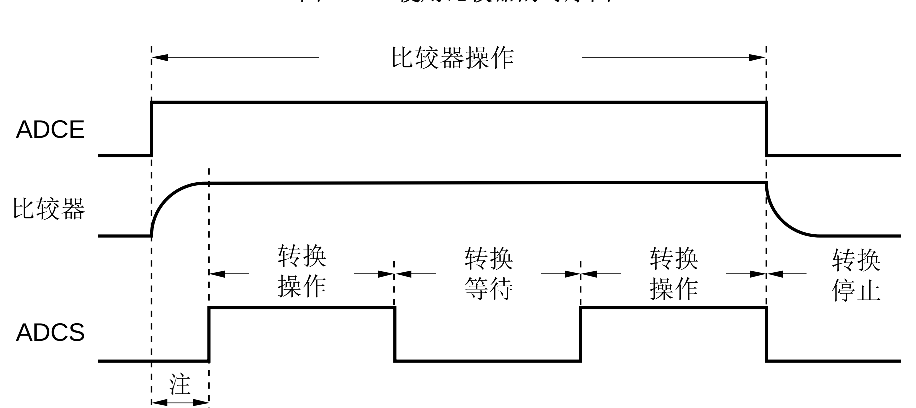
2. 比较器的的操作是由 ADCS 和 ADCE 控制的，从操作开始到操作稳定需要 1μs。因此，当 ADCE 被设置为 1 后，经过 1μs 或更长的时间后，再将 ADCS 置 1 时，此时的转换结果优于第一次的转换结果。否则，忽略第一次转换的数据。

表 12-1. ADCS 和 ADCE 的设置

ADCS	ADCE	A/D 转换操作
0	0	停止状态 （不存在直流功耗途径）
0	1	转换等待模式 （比较器操作，只有比较器产生功耗）
1	0	转换模式 （比较器停止操作[注]）
1	1	转换模式 （比较器操作）

注　忽略第 1 次转换的数据。

图 **12-4.** 使用比较器的时序图



注　　要稳定内部电路，从 ADCE 位的上升到 ADCS 位的下降时间间隔必须为 1 μs 或更长。

注意事项　**1.** 将 **FR0** 至 **FR3** 位、**LV1** 和 **LV0** 位重写为非原有数据之前，**A/D** 转换必须停止。

2. 如果有数据写入 **ADM**，会产生 **1** 个等待周期。当 **CPU** 运行于副系统时钟，且外设硬件时钟处于停止状态时，不要对 **ADM** 写入数据。详情参见第三十三章 等待注意事项。

表 12-2. A/D 转换时间的选择

（1） 2.7 V ≤ AV_{REF} ≤ 5.5 V

A/D 转换器模式寄存器（ADM）						转换时间的选择				转换时钟（f_{AD}）
FR3	FR2	FR1	FR0	LV1	LV0		f_{PRS} = 2 MHz	f_{PRS} = 8 MHz	f_{PRS} = 10 MHz	
1	×	×	×	0	0	352/f_{PRS}	禁止设置	44.0 μs	35.2 μs	f_{PRS}/16
0	0	0	0	0	0	264/f_{PRS}		33.0 μs	26.4 μs	f_{PRS}/12
0	0	0	1	0	0	176/f_{PRS}		22.0 μs	17.6 μs	f_{PRS}/8
0	0	1	0	0	0	132/f_{PRS}		16.5 μs	13.2 μs	f_{PRS}/6
0	0	1	1	0	0	88/f_{PRS}	44.0 μs	11.0 μs[注]	8.8 μs[注]	f_{PRS}/4
0	1	0	0	0	0	66/f_{PRS}	33.0 μs	8.3 μs[注]	6.6 μs[注]	f_{PRS}/3
0	1	0	1	0	0	44/f_{PRS}	22.0 μs	禁止设置	禁止设置	f_{PRS}/2
其他情况						禁止设置				

注　只有当 4.0 V ≤ AV_{REF} ≤ 5.5 V 时，才可以设置

（2） 2.3 V ≤ AV_{REF} < 2.7 V

A/D转换器模式寄存器（ADM）						转换时间的选择				转换时钟（f_{AD}）
FR3	FR2	FR1	FR0	LV1	LV0		f_{PRS} = 2 MHz	f_{PRS} = 5 MHz	f_{PRS} = 8 MHz	
0	0	0	0	0	1	480/f_{PRS}	禁止设置	禁止设置	60.0 μs	f_{PRS}/12
0	0	0	1	0	1	320/f_{PRS}		64.0 μs	40.0 μs	f_{PRS}/8
0	0	1	0	0	1	240/f_{PRS}		48.0 μs	30.0 μs	f_{PRS}/6
0	0	1	1	0	1	160/f_{PRS}		32.0 μs	禁止设置	f_{PRS}/4
0	1	0	0	0	1	120/f_{PRS}	60.0 μs	禁止设置		f_{PRS}/3
0	1	0	1	0	1	80/f_{PRS}	40.0 μs			f_{PRS}/2
其他情况						禁止设置				

<R>

注意事项 **1.** 根据以下条件设置转换时间

- **4.0 V ≤ AV_{REF} ≤ 5.5 V：　f_{AD} = 0.6** 至 **3.6 MHz**
- **2.7 V ≤ AV_{REF} < 4.0 V：f_{AD} = 0.6** 至 **1.8 MHz**
- **2.3 V ≤ AV_{REF} < 2.7 V：f_{AD} = 0.6** 至 **1.48 MHz**

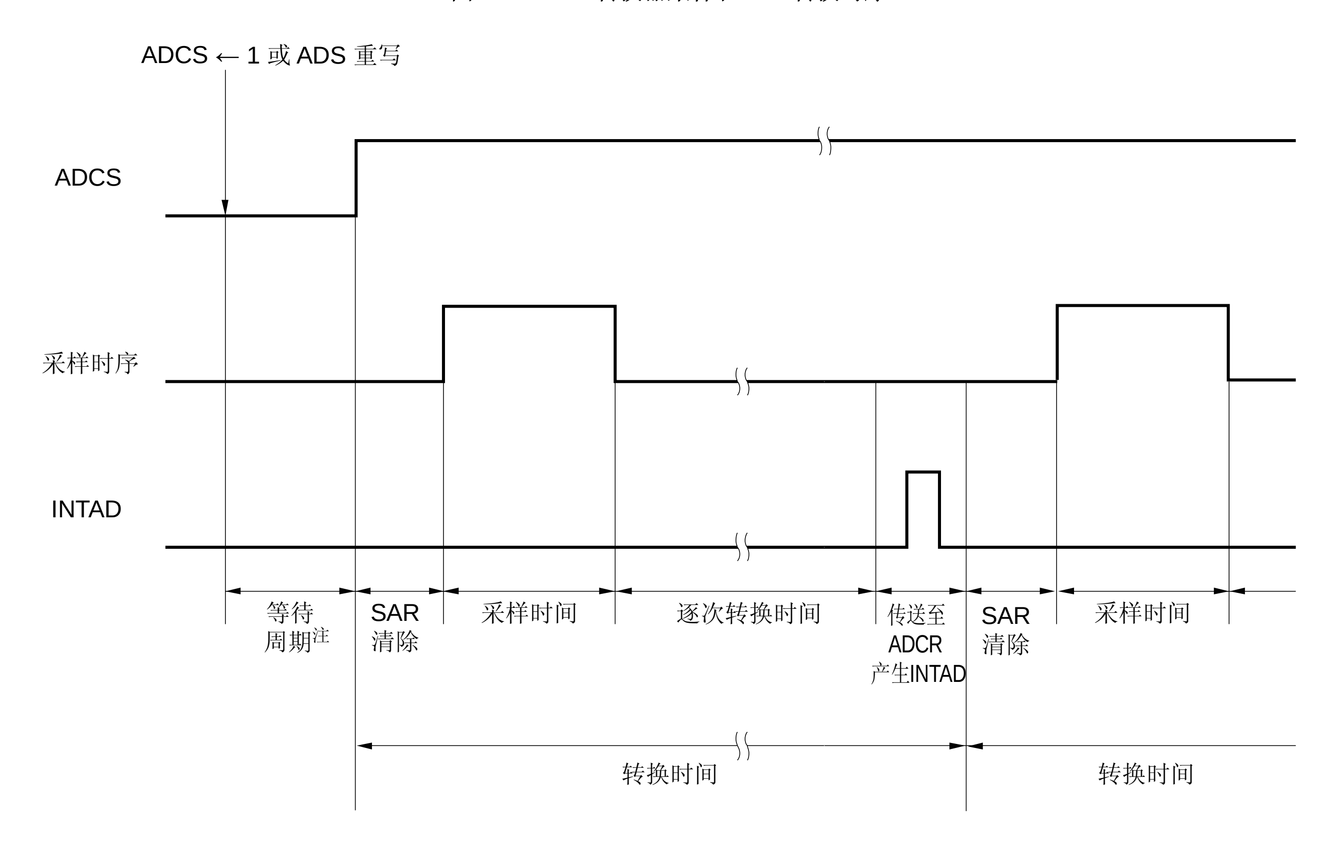
2. 将 **FR0** 至 **FR3** 位、**LV1** 和 **LV0** 位重写为非原有数据之前，要提前停止 **A/D** 转换一次（**ADCS = 0**）。

3. 当 **2.3 V ≤ AV_{REF} < 2.7 V** 时，修改 **LV1** 和 **LV0** 为非默认值。

4. 上述转换时间不包括时钟频率误差。在选择转换时间的时候，也要考虑时钟频率误差。

备注　f_{PRS}：　外设硬件时钟频率

图 12-5. A/D 转换器采样和 A/D 转换时序



注　　等待周期的详情参见第三十三章 等待注意事项。

（2）　10 位 A/D 转换结果寄存器（ADCR）

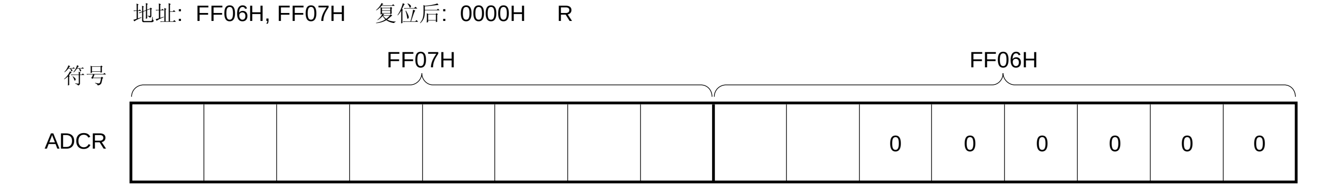
该寄存器是 16 位寄存器，用于存储 A/D 转换结果。寄存器的低 6 位固定为 0。每次 A/D 转换结束时，从逐次逼近寄存器装载转换结果。转换结果的高 8 位被存储在 FF07H，而转换结果的低 2 位存储在 FF06H 的高 2 位中。

可以通过 16 位存储器操作指令来读取 ADCR。

复位信号产生会将该寄存器清除为 0000H。

图 12-6. 10 位 A/D 转换结果寄存器（ADCR）的格式

地址: FF06H, FF07H　复位后: 0000H　R



符号	FF07H								FF06H							
ADCR											0	0	0	0	0	0

注意事项　**1.** 当写入 **A/D** 转换器模式寄存器（**ADM**）、模拟输入通道选择寄存器（**ADS**）和 **A/D** 端口配置寄存器（**ADPC0**）时，**ADCR** 的内容可能是不确定的。转换结束后，在写入 **ADM**、**ADS** 和 **ADPC0** 之前读取转换结果。如果没有使用上述时序，可能会读到不正确的转换结果。

2. 如果从 **ADCR** 读取数据，会产生一个等待周期。当 **CPU** 运行于副系统时钟，且外设硬件时钟处于停止状态时，不要从 **ADCR** 读取数据。详情参见第三十三章 等待注意事项。

（3） 8 位 A/D 转换结果寄存器（ADCRH）

该寄存器是 8 位寄存器，用于存储 A/D 转换结果。存储 10 位分辨率的高 8 位。。

可以通过 8 位存储器操作指令来读取 ADCRH。

复位信号产生会将该寄存器清除为 00H。

图 **12-7. 8** 位 **A/D** 转换结果寄存器（**ADCRH**）的格式

地址：FF07H　复位后：00H　R

符号	7	6	5	4	3	2	1	0
ADCRH								

注意事项 **1.** 当写入 **A/D** 转换器模式寄存器（**ADM**）、模拟输入通道选择寄存器（**ADS**）和 **A/D** 端口配置寄存器（**ADPC0**）时，**ADCRH** 的内容可能是不确定的。转换结束后，在写入 **ADM**、**ADS** 和 **ADPC0** 之前读取转换结果。如果没有使用上述时序，可能会读到不正确的转换结果。

2. 如果从 **ADCRH** 读取数据，会产生一个等待周期。当 **CPU** 运行于副系统时钟，且外设硬件时钟处于停止状态时，不要从 **ADCRH** 读取数据。详情参见第三十三章 等待注意事项。

（4） 模拟输入通道选择寄存器（ADS）

该寄存器用于选择将要进行 A/D 转换的模拟电压输入通道。

可以通过 1 位或 8 位存储器操作指令来设置 ADS。

复位信号产生会将该寄存器清除为 00H。

图 12-8. 模拟输入通道选择寄存器（ADS）

地址: FF8EH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ADS	0	0	0	0	0	ADS2	ADS1	ADS0

ADS2	ADS1	ADS0	模拟输入通道的选择
0	0	0	ANI0
0	0	1	ANI1
0	1	0	ANI2
0	1	1	ANI3
1	0	0	ANI4
1	0	1	ANI5
1	1	0	ANI6
1	1	1	ANI7

注意事项
1. 请确保将第 3 位至第 7 位清除为“0”。
2. 使用端口模式寄存器 2（PM2），将一个用于 A/D 转换的通道端口设置为输入模式。
3. 不要通过 ADPC0 和 ADS 将引脚设置用作数字 I/O。
4. 已经被选择作为 10 位逐次逼近型 A/D 转换器输入的通道引脚必须不能被选择作为 16 位ΔΣ型 A/D 转换器的输入。
5. 如果有数据写入 ADS，会产生一个等待周期。当 CPU 运行于副系统时钟，且外设硬件时钟处于停止状态时，不要对 ADS 写入数据。详情参见第三十三章 等待注意事项。

（5） A/D 端口配置寄存器 0（ADPC0）

该寄存器用于将 ANI0/P20 至 ANI7/P27 引脚切换为模拟输入（16 位ΔΣ型 A/D 转换器的模拟输入或 10 位逐次逼近型 A/D 转换器的模拟输入）或数字 I/O 端口。

可以通过 1 位或 8 位存储器操作指令来设置 ADPC0。

复位信号产生会将该寄存器清除为 08H。

图 12-9. A/D 端口配置寄存器 0（ADPC0）格式

地址：FF8FH　复位后：08H　R/W

符号	7	6	5	4	3	2	1	0
ADPC0	0	0	0	0	ADPC03	ADPC02	ADPC01	ADPC00

<μPD78F045x>

ADPC03	ADPC02	ADPC01	ADPC00	数字 I/O (D)/模拟输入 (A) 的切换							
				P27/ ANI7/ SEG32	P26/ ANI6/ SEG33	P25/ ANI5/ SEG34	P24/ ANI4/ SEG35	P23/ ANI3/ SEG36	P22/ ANI2/ SEG37	P21/ ANI1/ SEG38	P20/ ANI0/ SEG39
0	0	0	0	A	A	A	A	A	A	A	A
0	0	0	1	A	A	A	A	A	A	A	D
0	0	1	0	A	A	A	A	A	A	D	D
0	0	1	1	A	A	A	A	A	D	D	D
0	1	0	0	A	A	A	A	D	D	D	D
0	1	0	1	A	A	A	D	D	D	D	D
0	1	1	0	A	A	D	D	D	D	D	D
0	1	1	1	A	D	D	D	D	D	D	D
1	0	0	0	D	D	D	D	D	D	D	D
其它情况				禁止设置							

<μPD78F046x>

ADPC03	ADPC02	ADPC01	ADPC00	数字I/O (D)/模拟输入 (A: 逐次逼近型, Δ: ΔΣ 型) 的切换							
				P27/ ANI7/ REF+	P26/ ANI6/ REF-	P25/ ANI5/ DS2+	P24/ ANI4/ DS2-	P23/ ANI3/ DS1+	P22/ ANI2/ DS1-	P21/ ANI1/ DS0+	P20/ ANI0/ DS0-
0	0	0	0	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ
0	0	0	1	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A	D
0	0	1	0	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	D	D
0	0	1	1	A/Δ	A/Δ	A/Δ	A/Δ	A	D	D	D
0	1	0	0	A/Δ	A/Δ	A/Δ	A/Δ	D	D	D	D
0	1	0	1	A	A	A	D	D	D	D	D
0	1	1	0	A	A	D	D	D	D	D	D
0	1	1	1	A	D	D	D	D	D	D	D
1	0	0	0	D	D	D	D	D	D	D	D
其它情况				禁止设置							

注意事项
1. 使用端口模式寄存器 2（PM2），将一个用于 A/D 转换的通道端口设置为输入模式。
2. 不要通过 ADS、ADDS1 或 ADDS0 将 ADPC0 设置的引脚用作数字 I/O。
3. 如果有数据写入 ADPC0，会产生一个等待周期。当 CPU 运行于副系统时钟，且外设硬件时钟处于停止状态时，不要对 ADPC0 写入数据。详情参见第三十三章 等待注意事项。
4. 如果通过 PF2 寄存器将 ANI0/P20/SEG31 至 ANI7/P27/SEG24 引脚设置为 segment 输出引脚，不论 ADPC0 如何设置，输出都被设置为 segment 输出（仅适用于 μPD78F045x）。

（6） 端口模式寄存器 2（PM2）

使用 ANI0/P20 至 ANI7/P27 引脚作为模拟输入端口时，将 PM20 至 PM27 设置为 1。此时，P20 至 P27 的输出锁存器的内容可能为 0 或 1。

如果 PM20 至 PM27 被设置为 0，则它们不能被用作模拟输入端口引脚。

可以通过 1 位或 8 位存储器操作指令来设置 PM2。

复位信号产生会将该寄存器清除为 FFH。

图 12-10. 端口模式寄存器 2（PM2）的格式

地址: FF22H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20

PM2n	P2n 引脚I/O 模式的选择 (n = 0 至 7)
0	输出模式 (输出缓冲器打开)
1	输入模式 (输出缓冲器关闭)

根据 PF2、ADPC0、PM2、ADS 和 ADDCTL0 的设置，ANI0/P20 至 ANI7/P27 引脚如下。

表 12-3. ANI0/P20 至 ANI7/P27 引脚的功能设置

（a） μPD78F045x

PF2	ADPC0	PM2	ADS	P20/SEG31/ANI0 至 P27/SEG24/ANI7引脚
数字/模拟选择	模拟输入的选择	输入模式	不选择ANI	模拟输入 （不被转换）
			选择ANI	模拟输入（通过逐次逼近型A/D转换器进行转换）
		输出模式	–	禁止设置
	数字I/O的选择	输入模式	–	数字输入
		输出模式	–	数字输出
SEG 输出选择	–	–	–	Segment输出

（b） μPD78F046x

ADPC0	PM2	ADS	ADDCTL0	P20/ANI0/DS0- 至P27/ANI7/REF+引脚
模拟输入选择	输入模式	不选择ANI.	不选择DSn±.	模拟输入 （不被转换）
		选择ANI.	不选择DSn±.	模拟输入（通过逐次逼近型A/D转换器进行转换）
		不选择ANI.	选择DSn±.	模拟输入 （通过ΔΣ型A/D转换器进行转换）
		选择ANI.	选择DSn±.	禁止设置
	输出模式	–		禁止设置
数字I/O 选择	输入模式	–		数字输入
	输出模式	–		数字输出

12.4 10 位逐次逼近型A/D转换器的操作

12.4.1 A/D转换器的基本操作

<1> 把 A/D 转换器模式寄存器的第 0 位（ADCE）置 1，可以启动比较器的操作。

<2> 通过 A/D 端口配置寄存器（ADPC0）设置对模拟输入进行 A/D 转换的通道，并且通过端口模式寄存器 2（PM2）设置为输入模式。

<3> 使用 ADM 的第 6 位至第 1 位（FR3 至 FR0、LV1 和 LV0）设置 A/D 转换时间。

<4> 使用模拟输入通道选择寄存器（ADS）指定一个通道用于 A/D 转换。

<5> 设置 ADM 的第 7 位（ADCS）为 1，启动转换操作。
（<6>至<12> 是由硬件完成的操作。）

<6> 通过采样&保持电路对输入到被选中模拟输入通道的电压进行采样。

<7> 在经过一段时间的采样后，采样&保持电路处于保持状态，且在 A/D 转换操作结束之前一直保持采样电压。

<8> 设置逐次逼近寄存器（SAR）的第 9 位。通过档位选择器将串联电阻串的电压档位设置为（1/2） AV_{REF}。

<9> 通过电压比较器对串联电阻串的电压档位和采样电压进行比较。如果模拟输入电压高于（1/2） AV_{REF}，则 SAR 的 MSB 被设置为 1；如果模拟输入电压低于（1/2） AV_{REF}，则 MSB 被重置为 0。

<10> 接下来，SAR 的第 8 位自动置 1，并进入下一个比较操作。根据第 9 位的预置值选择串联电阻串的电压档位，描述如下。
- 第 9 位 = 1： （3/4） AV_{REF}
- 第 9 位 = 0： （1/4） AV_{REF}

比较电压档位和采样电压，且 SAR 的第 8 位的操作如下。
- 模拟输入电压 ≥ 电压档位：第 8 位 = 1
- 模拟输入电压 < 电压档位：第 8 位 = 0

<11> 按此方式继续进行比较，直到 SAR 的第 0 位。

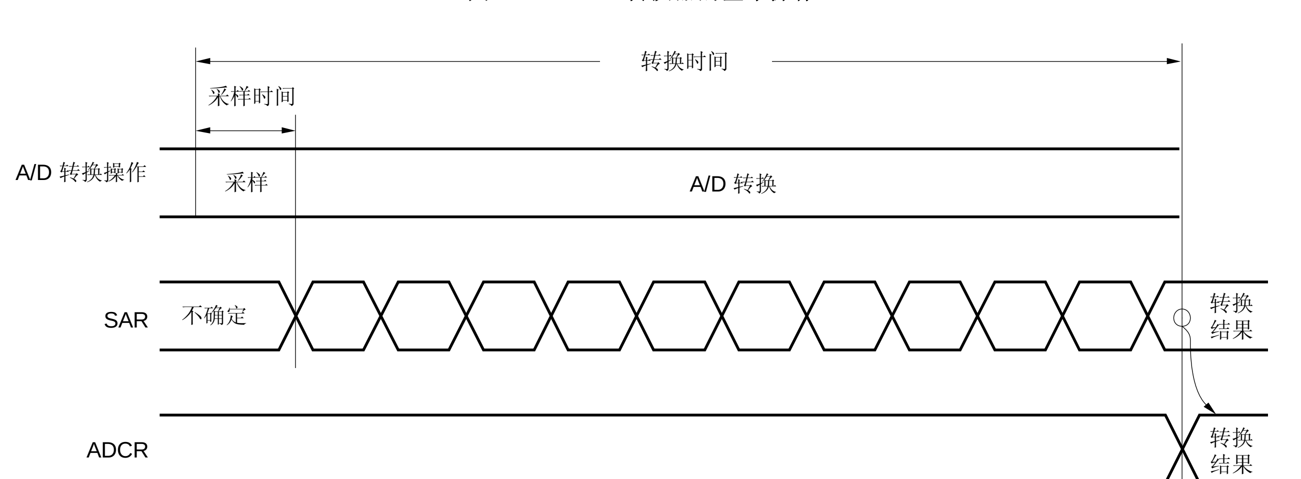
<12> 在 10 位比较完成后，在 SAR 中保留一个有效数字结果，然后结果值被传送到 A/D 转换结果寄存器（ADCR，ADCRH）中，并被锁存。
同时，也会产生 A/D 转换结束中断请求（INTAD）。

<13> 反复步骤 <6>至<12>，直至 ADCS 被清除为 0。
要停止 A/D 转换器操作，将 ADCS 清除为 0。
当 ADCE = 1 时，若要重新启动 A/D 转换，应从步骤<5>开始。当 ADCE = 0 时，若要再次启动 A/D 转换，设置 ADCE=1，等待 1 μs 或更久，然后从步骤<5>开始。如要改变 A/D 转换的通道，则从步骤<4>开始。

注意事项 必须确保 <1>至<5> 的操作时间为 1 μs 或更久。

备注 有两种类型的 A/D 转换结果寄存器可供使用。
- ADCR （16 位）： 存储 10 位 A/D 转换值
- ADCRH （8 位）： 存储 8 位 A/D 转换值

图 12-11. A/D 转换器的基本操作



连续执行 A/D 转换操作，直到通过软件将 A/D 转换器模式寄存器（ADM）的第 7 位（ADCS）重置（0）。

在 A/D 转换期间，如果对模拟输入通道选择寄存器（ADS）执行写入操作，则转换操作被初始化，并且如果 ADCS 被置位（1），则转换操作再次重头开始。

复位信号产生会将 A/D 转换结果寄存器（ADCR）的内容清除为 0000H 或 00H。

12.4.2 输入电压和转换结果

输入到模拟输入引脚（ANI0 至 ANI7）的模拟输入电压与理论 A/D 转换结果（存储在 10 位 A/D 转换结果寄存器（ADCR））之间的关系表示如下。

$$SAR = INT\left(\frac{V_{AIN}}{AV_{REF}} \times 1024 + 0.5\right)$$

$$ADCR = SAR \times 64$$

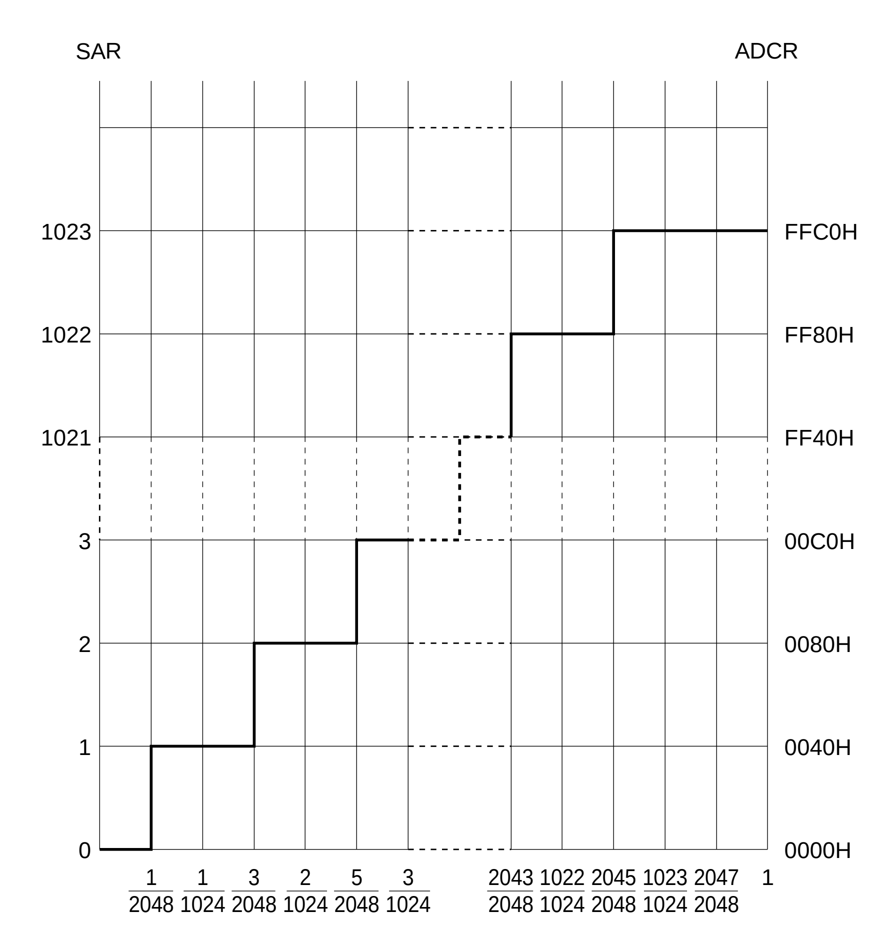
或

$$\left(\frac{ADCR}{64} - 0.5\right) \times \frac{AV_{REF}}{1024} \leq V_{AIN} < \left(\frac{ADCR}{64} + 0.5\right) \times \frac{AV_{REF}}{1024}$$

其中，INT（）：　该函数返回括号中值的整数部分。
V_{AIN}：　模拟输入电压
AV_{REF}：　AV_{REF} 引脚电压
ADCR：　A/D 转换结果寄存器（ADCR）的值
SAR：　逐次逼近寄存器

图 12-12 显示模拟输入电压与 A/D 转换结果之间的关系。

图 12-12. 模拟输入电压和 A/D 转换结果之间的关系



12.4.3 A/D 转换器操作模式

A/D 转换器的操作模式是选择模式。通过模拟输入通道规范寄存器（ADS）从 ANI0 至 ANI7 中选择一个模拟输入的通道，并执行 A/D 转换。

（1） A/D 转换操作

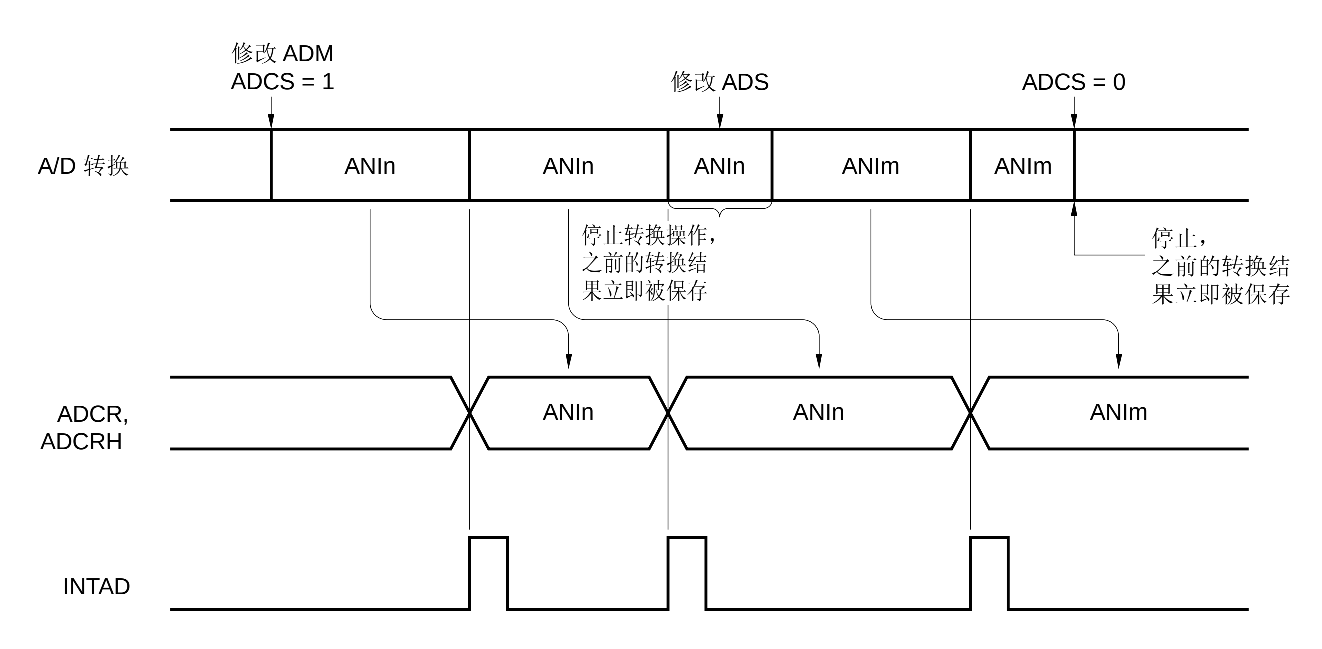
将 A/D 转换器模式寄存器（ADM）的第 7 位（ADCS）设置为 1，可以启动电压的 A/D 转换操作，该电压由模拟输入通道选择寄存器（ADS）指定的模拟输入引脚输入。

当 A/D 转换完成时，A/D 转换结果存储在 A/D 转换结果寄存器（ADCR）中，并产生一个中断请求信号（INTAD）。当一个 A/D 转换结束时，下一个 A/D 转换操作立即开始。

如果在 A/D 转换期间 ADS 的值被重写，则停止当前正在执行的 A/D 转换操作，重头启动。

如果在 A/D 转换期间 ADCS 被置 0，则 A/D 转换立即停止。此时，之前的瞬时转换结果被保存。

图 12-13. A/D 转换操作



备注 **1.** n = 0 至 7
2. m = 0 至 7

设置方法描述如下。

<1> 设置 A/D 转换器模式寄存器（ADM）的第 0 位（ADCE）为 1。
<2> 使用 A/D 端口配置寄存器的第 3 位至第 0 位（ADPC03 至 ADPC00）和端口模式寄存器 2（PM2）的第 7 位至第 0 位（PM27 至 PM20），设置在模拟输入模式下使用的通道。
<3> 根据 ADM 的第 6 位至第 1 位（FR3 至 FR0、LV1 和 LV0）来选择转换时间。
<4> 使用模拟输入通道规范寄存器（ADS）的第 2 位至第 0 位（ADS2 至 ADS0）来选择通道。
<5> 将 ADM 的第 7 位（ADCS）置 1，启动 A/D 转换操作。
<6> 当一次 A/D 转换结束时，产生一个中断请求信号（INTAD）。
<7> A/D 转换数据被传送到 A/D 转换结果寄存器（ADCR，ADCRH）中。

<改变通道>

<8> 使用 ADS 的第 2 位至第 0 位（ADS2 至 ADS0）改变通道，启动 A/D 转换操作。
<9> 当一次 A/D 转换结束时，产生一个中断请求信号（INTAD）。
<10> A/D 转换数据被传送到 A/D 转换结果寄存器（ADCR，ADCRH）中。

<完成 A/D 转换>

<11> 将 ADCS 清除为 0。
<12> 将 ADCE 清除为 0。

注意事项 **1. 必须确保步骤 <1>至<5>的操作时间为 1 μs 或更久。**
2. 步骤<1> 可以在<2> 和 <4>之间进行。
3. 步骤<1>可以省略。但是在这种情况下，步骤<5>之后的第一个转换结果被忽略。
4. 根据 ADM 的第 6 位至第 1 位（FR3 至 FR0、LV1、LV0）设置的转换时间和从步骤<6>到步骤<9>的操作时间不同。步骤<8>至<9>的操作时间是由 FR3 至 FR0、LV1 和 LV0 设置的转换时间。

12.5 逐次逼近型A/D转换器特征表的阅读方法

这里解释逐次逼近型 A/D 转换器特有的专用术语。

（1） 分辨率

这是可以识别的最小的模拟输入电压，即数字输出的每位占模拟输入电压的百分比，称为 1LSB（最低有效位）。1 LSB 对满量程的比率被表示为%FSR（满标度量程）。

当分辨率为 10 位时 1LSB 表示如下。

$$1LSB = 1/2^{10} = 1/1024$$
$$= 0.098\%FSR$$

精确度与分辨率无关，而由总误差决定。

（2） 总误差

总误差是指实际测量值与理论值之间的最大误差。
零标度误差、满量程误差、积分线性误差和微分线性误差等组合起来表示总误差。
注意量化误差不属于特征表中总误差的范围。

（3） 量化误差

当模拟值转换为数字值时，会不可避免的产生±1/2 LSB 的误差。在一个 A/D 转换器中，相差±1/2LSB 的模拟输入电压被转换为相同的数字值，所以量化误差不可避免。
注意量化误差不属于特征表中的总误差、零标度误差、满量程误差、积分线性误差和微分线性误差。

图 12-14. 总误差

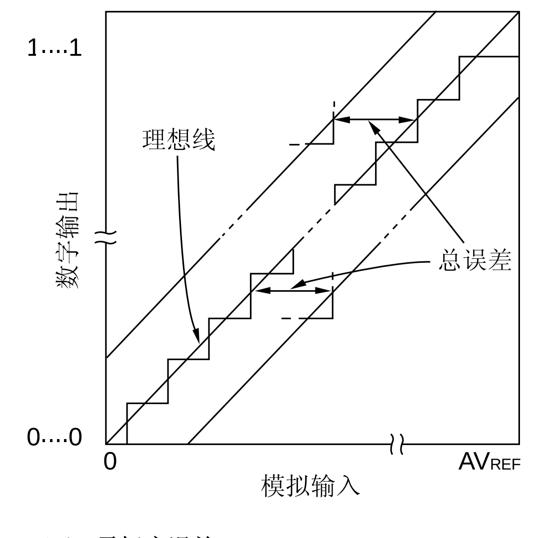
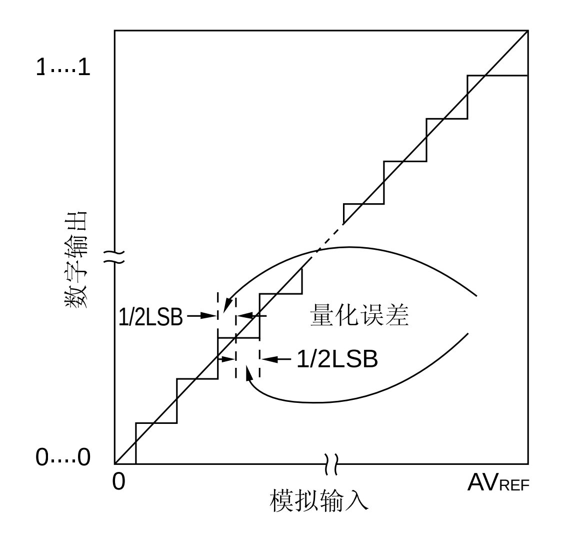


图 12-15. 量化误差



（4） 零标度误差

零标度误差表示当数字输出从 0......000 改变为 0......001 时模拟输入电压的实际测量值与理论值（1/2LSB）之间的误差。
如果实际测量值大于理论值，当数字输出范围在 0……001 至 0……010 之间时，零标度误差表示模拟输入电压的实际测量值与理论值（3/2LSB）之间的差别。

（5） 满量程误差

当数字输出从 1......110 改变为 1......111 时，满量程误差表示模拟输入电压的实际测量值与理论值（满量程 – 3/2LSB）之间的差别。

（6） 积分线性误差

积分线性误差表示转换特征偏离理想线性关系的程度。它表示当零标度误差和满量程误差均为 0 时，实际测量值与理想直线之间误差的最大值。

（7） 微分线性误差

当代码输出的理想宽度为 1LSB 时，微分线性误差表示实际测量值与理想值之间的差别。

图 12-16. 零标度误差

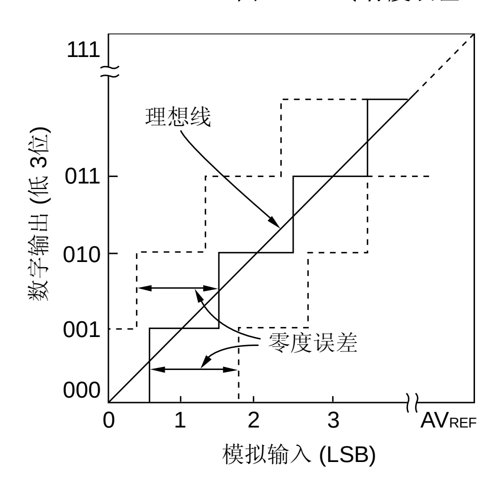


图 12-17. 满量程误差

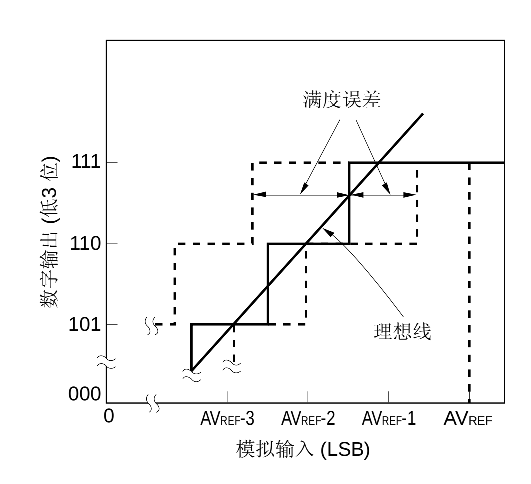


图 12-18. 积分线性误差

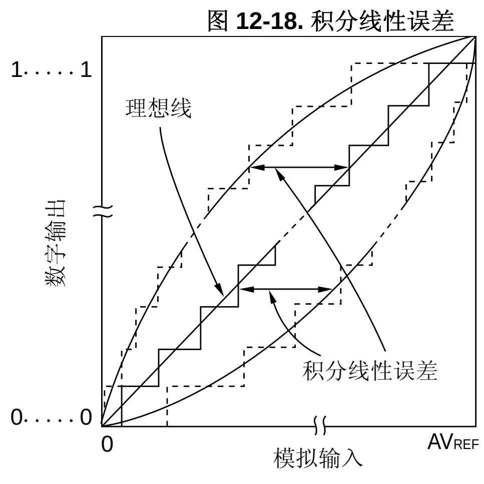
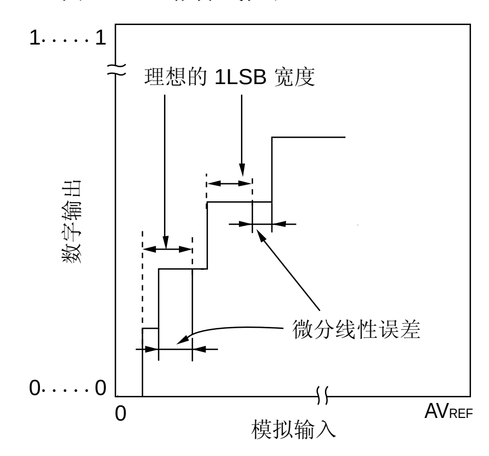


图 12-19. 微分线性误差



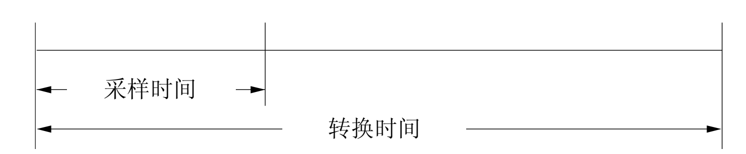
（8） 转换时间

转换时间表示从开始采样到获取数字输出所经历的时间。

采样时间包含在特征表的转换时间中。

（9） 采样时间

采样时间表示采样所需的时间，从为模拟电压打开模拟开关，到模拟电压被采样&保持电路采样。



12.6 A/D转换器使用注意事项

（1） STOP 模式下的操作电流

在 STOP 模式下，A/D 转换器停止操作。此时，将 A/D 转换器模式寄存器（ADM）的第 7 位（ADCS）和第 0 位（ADCE）清除为 0，可以降低操作电流。

若要从待机状态重新启动，将中断请求标志寄存器 1L （IF1L）的第 0 位（ADIF）清除为 0，然后开始操作。

（2） ANI0 至 ANI7 的输入范围

注意 ANI0 至 ANI7 输入电压的额定范围。如果输入到模拟输入通道的电压大于等于 AV_{REF}，或者小于等于 AV_{SS}（即使在绝对最大额定范围之内），则该通道的转换值不确定。此外，其他通道的转换值也可能会受影响。

（3） 冲突操作

<1> 在转换结束时，对 A/D 转换结果寄存器（ADCR，ADCRH）的写入操作和通过指令读取 ADCR 或 ADCRH 的操作之间的冲突。
ADCR 或 ADCRH 读操作优先。在执行读操作之后，新的转换结果被写入 ADCR 或 ADCRH。

<2> 在转换结束时，对 ADCR 或 ADCRH 的写入操作和对 A/D 转换器模式寄存器（ADM）的写入操作、对模拟输入通道规范寄存器（ADS）或对 A/D 端口配置寄存器（ADPC0）的写入操作之间的冲突。
ADM、ADS 或 ADPC0 的写入操作优先。不执行对 ADCR 或 ADCRH 的写入，也不会产生转换结束中断信号（INTAD）。

（4） 解决噪声问题的方法

为了保证 10 位分辨率，必须注意输入到 AV_{REF} 引脚和 ANI0 至 ANI7 引脚的噪声。

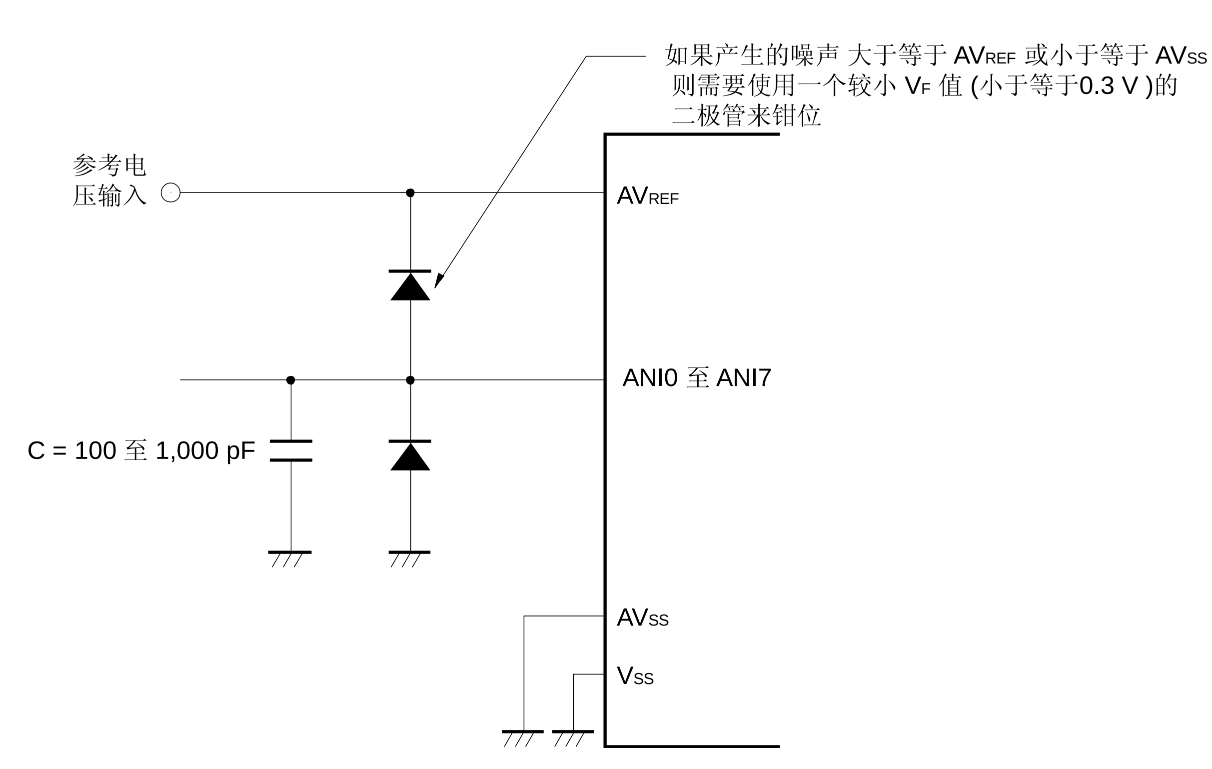
<1> 在电源上连接一个具有较低等效电阻和优良频率响应的电容。

<2> 模拟输入源的输出阻抗越大，影响就越大。为了降低噪声，推荐按照图 13-20 所示连接外部 C。

<3> 在转换期间不要切换引脚。

<4> 如果在转换启动后立即设置 HALT 模式，则可以改善精度。

图 12-20. 模拟输入引脚连接



（5） ANI0/SEG31/P20 至 ANI7/SEG24/P27 引脚 （*u*PD78F045x），
ANI0/DS0–/P20 至 ANI7/REF+/P27 引脚 （*u*PD78F046x）

<1> 模拟输入引脚（ANI0 至 ANI7）也可以用作输入端口引脚（P20 至 P27）。
当通过 ANI0 至 ANI7 中的任意一个选中的通道执行 A/D 转换时，转换过程进行中不要访问 P20 至 P27；否则转换分辨率可能会降低。推荐从离 AVREF 最远的 ANI0/P20 开始，选择 P20 至 P27 的引脚作为数字 I/O 端口。

<2> 如果正在进行 A/D 转换的引脚的相邻引脚上数字脉冲被输入或输出，或 segment-输出，则由于噪声耦合，有可能得不到预期的 A/D 转换值。因此，在进行 A/D 转换时，不要在相邻引脚输入或输出脉冲或 segment 输出。

（6） ANI0 至 ANI7 引脚的输入阻抗

采样期间 A/D 转换器对采样电容充电，以便采样。
因此，不进行采样时仅有漏电流流过，而在采样期间则有电容充电的电流流过，因此，输入阻抗的波动会根据是否在进行采样或位于其他状态。
为了确保有效采样，推荐保持模拟输入源的输出阻抗小于等于 10 kΩ，并将一个 100 pF 左右的电容连接到 ANI0 至 ANI7 引脚（参见 **图 12-20**）。

（7） AVREF 引脚输入阻抗

在 AVREF 与 AVSS 引脚之间连接几十千欧的串联电阻串。
因此，如果参考电压源的输出阻抗很大，当它串联连接到 AVREF 与 AVSS 引脚之间的串联电阻串时，会导致较大的参考电压误差。

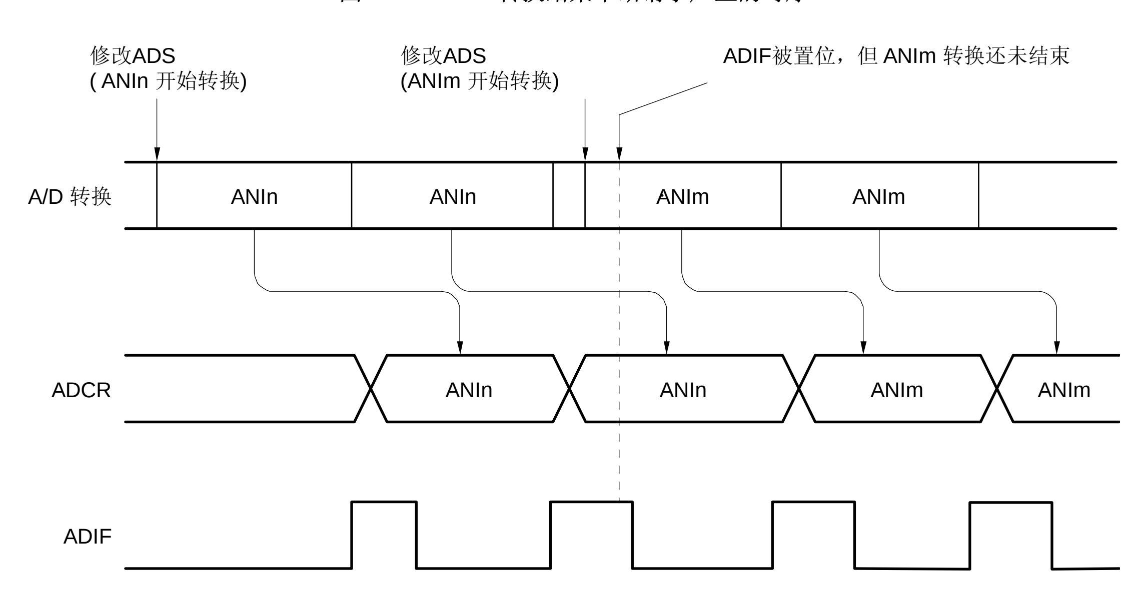
（8） 中断请求标志（ADIF）

即使模拟输入通道选择寄存器（ADS）被改变，中断请求标志（ADIF）也不会被清除。

因此，如果在 A/D 转换期间有一个模拟输入引脚被改变，则就在 ADS 被修改之前，改变前的模拟通道的 A/D 转换结果和 ADIF 可能刚被设置。因此需要特别注意，此时，当 ADS 重写后立即读取 ADIF 时，即使修改后的模拟输入的实际 A/D 转换尚未结束，也会设置 ADIF。

当 A/D 转换停止后又重新恢复时，在 A/D 转换操作被恢复前，清除 ADIF。

图 12-21. A/D 转换结束中断请求产生的时序



备注 **1.** n = 0 至 7
2. m = 0 至 7

（9） **A/D 转换刚启动后的转换结果**

若在 ADCE 置 1 后的 1 μs 内对 ADCS 置 1，或在 ADCE=0 时对 ADCS 置 1，那么 A/D 转换刚启动后第一次 A/D 转换值可能不在额定范围内。可以采取措施，比如轮询 A/D 转换结束中断请求（INTAD），并删除第 1 次转换结果。

（10） **A/D 转换结果寄存器（ADCR，ADCRH）的读取操作**

对 A/D 转换器模式寄存器（ADM）、模拟输入通道选择寄存器（ADS）和 A/D 端口配置寄存器 0（ADPC0）进行写入操作时，ADCR 和 ADCRH 的内容可能是不确定的。转换结束后，应该在对 ADM、ADS 和 ADPC0 进行写入操作之前读取转换结果。如果采用的时序与上述不同，可能会导致读取到的转换结果不正确。

<R> **（11） 内部等效电路**

模拟输入模块的等效电路如下所示。

图 12-22 ANIn 引脚的内部等效电路

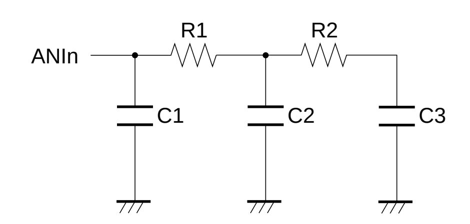


表 12-4. 等效电路的电阻和电容值（参考值）

AV_{REF}	R1	C1	C2
4.0 V ≤ AV_{REF} ≤ 5.5 V	8.1 kΩ	8 pF	5 pF
2.7 V ≤ AV_{REF} < 4.0 V	31 kΩ	8 pF	5 pF
2.3 V ≤ AV_{REF} < 2.7 V	381 kΩ	8 pF	5 pF

备注 **1.** 表 12-4 列出的电阻和电容值不是保证值。
2. n = 0 至 7

（12） 同时使用 10 位逐次逼近型 A/D 转换器和 16 位ΔΣ型 A/D 转换器（仅限 PD78F046x）

同时使用 10 位逐次逼近型 A/D 转换器和 16 位ΔΣ型 A/D 转换器时，A/D 转换器的精度会降低。

当 10 位逐次逼近型 A/D 转换器操作期间，应该停止 16 位ΔΣ型 A/D 转换器，否则将无法保证操作精度。同样，当 16 位ΔΣ型 A/D 转换器操作期间，应该停止 10 位逐次逼近型 A/D 转换器。（不要同时操作）

13.1 16 位ΔΣ类型A/D转换器的功能

16 位ΔΣ型 A/D 转换器用于将模拟输入信号转换为数字信号，最多包括 3 个通道（DS0–/DS0+，DS1–/DS1+，DS2–/DS2+），具有 16 位分辨率。

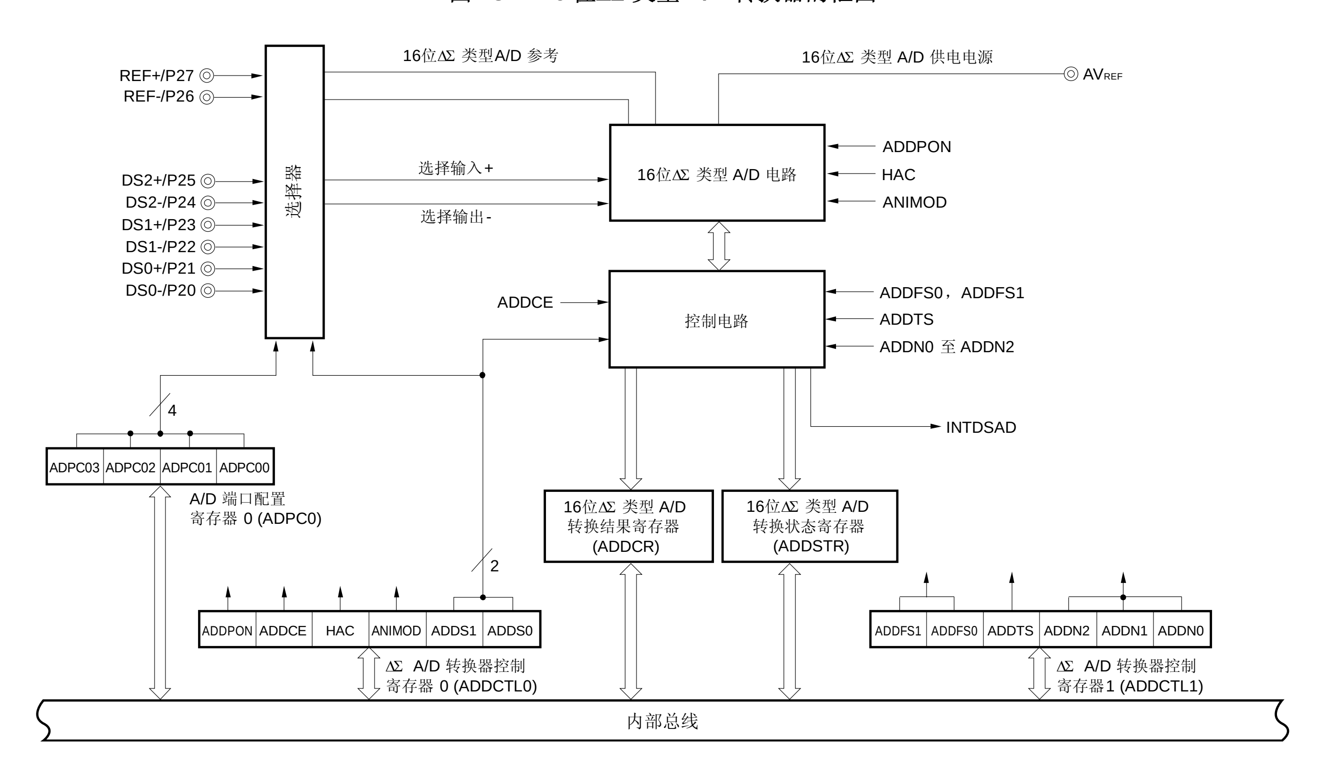
A/D 转换器有以下功能。

- **16 位分辨率 A/D 转换**

  从 DS0–/DS0+，DS1–/DS1+或 DS2–/DS2+中选择一个模拟输入通道，对其重复执行 16 位分辨率 A/D 转换。每次 A/D 转换结束，都一个中断请求（INTDSAD）。

  通过降低分辨率可以缩短转换时间。

图 13-1. 16 位ΔΣ 类型 A/D 转换器的框图



13.2 16 位ΔΣ类型A/D转换器的配置

16 位ΔΣ类型 A/D 转换器包含下列硬件。

（1） DS0–/DS0+，DS1–/DS1+和 DS2–/DS2+引脚

16 位ΔΣ类型 A/D 转换器共有 3 通道模拟输入引脚。从这些引脚输入的模拟信号被转变为数字信号。除了那个被选为模拟输入的引脚外，其他引脚均可用作 I/O 端口引脚。

在这些引脚用于差分输入模式时，输入模拟信号到 DS–和 DS+。这些引脚用于单信号输入模式时，将模拟信号输入到 DS+引脚，并设定 DS–引脚的电位与 V_{SS} 和 AV_{SS} 相同。

（2） 16 位ΔΣ类型 A/D 电路

16 位ΔΣ型 A/D 电路根据参考电压将采样到的电压值转换成为数字值，并将其输出到控制电路。

（3） 控制电路

控制电路控制转换的时间，也控制要进行 A/D 转换的模拟输入转换操作的开始/停止。当 A/D 转换完成时，转换结果被传输到 16 位ΔΣ型 A/D 转换结果寄存器（ADDCR），并产生中断 INTDSAD。

（4） 16 位ΔΣ型 A/D 转换结果寄存器（ADDCR）

每当 A/D 转换完成时，转换结果都会从控制电路装载到该寄存器，ADDCR 寄存器将 A/D 转换结果保存在高 16 位。

（5） 8 位ΔΣ型 A/D 转换结果寄存器（ADDCRH）

每当 A/D 转换完成时，转换结果都会从控制电路装载到该寄存器，ADDCRH 寄存器存储 A/D 转换结果的高 8 位。

注意事项 **从 ADDCR 和 ADDCRH 读取数据时，会产生一个等待周期。当 CPU 运行于副系统时钟，且外设硬件时钟处于停止状态时，不要从 ADDCR 和 ADDCRH 读取数据。**

（6） AV_{REF} 引脚

该引脚为 16 位ΔΣ型 A/D 电路输入一个模拟电源。当端口 2 有至少一个端口被用作数字端口时或用作 segment 输出时，应该设置该引脚与 V_{DD} 的电位相同。

（7） REF– 和 REF+引脚

该引脚为 16 位ΔΣ型 A/D 转换器输入参考电压。输入到 DS0–/DS0+，DS1–/DS1 和 DS2–/DS2+的信号会根据 REF–和 REF+之间的参考电压转换为数字信号。REF–和 REF+引脚必须分别与 V_{SS}/AV_{SS} 和 AV_{REF} 引脚保持相同的电位。

（8） AV_{SS} 引脚

这是 A/D 转换器的地电位引脚。即便没有使用 A/D 转换器，该引脚的电压始终应该和 V_{SS} 引脚保持相同的电位。

（9） 16 位ΔΣ型 A/D 转换器控制寄存器 0（ADDCTL0）

该寄存器用于设置 16 位ΔΣ类型 A/D 电路或控制电路的上电/下电状态，转换开始/停止状态，高精度模式开/关状态，ΔΣ输入模式控制和模拟输入通道。

（10） 16 位ΔΣ型 A/D 转换器控制寄存器 1（ADDCTL1）

该寄存器用于设置 A/D 转换使用的采样时钟，串行/并行模式状态和采样计数（分辨率）。

（11） 16 位ΔΣ型 A/D 转换状态寄存器（ADDSTR）

当 16 位ΔΣ型 A/D 转换操作完成（产生转换完成中断），同时发生转换通道改变时，该寄存器用于检查哪个通道已经完成转换。

（12） A/D 端口配置寄存器 0（ADPC0）

该寄存器用于将 ANI0/P20/DS0–至 ANI7/P27/REF+ 引脚切换为模拟输入（16 位ΔΣ型 A/D 转换器的模拟输入或 10 位逐次逼近型 A/D 转换器的模拟输入）或端口的数字 I/O。

（13） 端口模式寄存器 2（PM2）

该寄存器将 ANI0/P20/DS0–至 ANI7/P27/REF+ 引脚切换为输入或输出。

13.3 16 位ΔΣ型A/D转换器使用的寄存器

16 位ΔΣ 型 A/D 转换器使用下列 9 个寄存器

- 16 位 ΔΣ 型 A/D 转换器控制寄存器 0（ADDCTL0）
- 16 位 ΔΣ 型 A/D 转换器控制寄存器 1（ADDCTL1）
- 16 位 ΔΣ 型 A/D 转换结果寄存器 （ADDCR）
- 8 位 ΔΣ 型 A/D 转换结果寄存器 （ADDCRH）
- 16 位 ΔΣ 型 A/D 转换状态寄存器 （ADDSTR）
- A/D 端口配置寄存器 0（ADPC0）
- 16 位 ΔΣ 型 A/D 采样延迟时间设置使能寄存器
- 16 位 ΔΣ 型 A/D 采样延迟时间设置寄存器
- 端口模式寄存器 2（PM2）

（1）　16 位ΔΣ 型 A/D 转换器控制寄存器 0（ADDCTL0）

该寄存器用于设置 16 位ΔΣ类型 A/D 电路或控制电路的上电/下电状态，转换开始/停止状态，高精度模式开/关状态，ΔΣ输入模式控制和模拟输入通道

可以通过 1 位或 8 位存储器操作指令来设置 ADDCTL0 。

复位信号产生会将该寄存器清除为 00H。

图 13-2. 16 位 ΔΣ 型 A/D 转换器控制寄存器 0（ADDCTL0）的格式

地址： FF7CH 复位后： 00H R/W

符号	<7>	<6>	<5>	<4>	3	2	1	0
ADDCTL0	ADPON	ADDCE	HAC	AINMCD	0	0	ADDS1	ADDS0

ADDPON	16位 ΔΣ 型 A/D 电路电源控制
0	电源关闭
1	电源打开

ADDCE	16位 ΔΣ 型 A/D 转换操作控制
0	停止转换操作
1	开始转换操作

HAC	设置 16位 ΔΣ 型 A/D 转换高精确度模式
0	高精确度模式关闭
1	高精确度模式打开

AINMOD	16位 ΔΣ 类型 A/D 转换输入模式控制
0	单信号输入
1	差分输入

ADDS1	ADDS0	16位 ΔΣ 型 模拟输入设定
0	0	DS0+/DS0-
0	1	DS1+/DS1-
1	0	DS2+/DS2-
1	1	禁止设置

注意事项
1. 不要将 **ADDPON** 和 **ADDCE** 同时设置为 **1**。在 **ADDPON** 设置为 **1** 后，必须经过至少 **1.2μs** 后，才能将 **ADDCE** 设置为 **1**。
2. 如果某个引脚已经由 **AFPC0** 寄存器选择作为模拟输入模式，那么不能再通过 **ADDS1** 和 **ADDS0** 设定其作为ΔΣ模拟输入通道。
3. 禁止同时运行 **16** 位 ΔΣ 型 **A/D** 转换和 **10** 位逐次逼近型 **A/D** 转换（**ADDCE = 1** 且 **ADCS = 1**）。
4. 如果 **ADDCTL0** 被重写（包含相同的数值），那么必须将 **A/D** 重新初始化才能恢复转换操作。
5. 输入电压的设置要按照表 **13-4** 输入电压范围。
6. 当执行 **STOP** 指令时，**16** 位 ΔΣ 型 **A/D** 转换器的供电必须关闭（**ADDPON = 0**）。

（2） 16 位 ΔΣ 型 A/D 转换器控制寄存器 1 （ADDCTL1）

该寄存器用于设置 A/D 转换使用的采样时钟，串行/并行模式状态和采样计数（分辨率）。

可以通过 1 位或 8 位存储器操作指令来设置 ADDCTL1 。

复位信号产生会将该寄存器清除为 00H。

图 13-3. 16 位ΔΣ型 A/D 转换器控制寄存器 1（ADDCTL1）的格式

地址：FF7DH 复位后：00H R/W

符号	7	6	<5>	4	3	2	1	0
ADDCTL1	ADDFS1	ADDFS0	ADDTS	0	0	ADDN2	ADDN1	ADDN0

ADDFS1	ADDFS0	16位 ΔΣ 型 A/D 采样时钟 (f_{VP}) 选择
0	0	$f_{PRS}/4$
0	1	$f_{PRS}/8$
1	0	$f_{PRS}/16$
1	1	$f_{SUB}/2$

ADDTS	设置 16位ΔΣ 型 A/D 串行/并行模式
0	串行模式
1	并行模式

ADDN2	ADDN1	ADDN0	16位 ΔΣ 型 A/D 采样N的数值（分辨率）
0	0	0	256 (8 位)
0	0	1	1024 (10 位)
0	1	0	2048 (11 位)
0	1	1	4096 (12 位)
1	0	0	8192 (13 位)
1	0	1	16384 (14 位)
1	1	0	32768 (15 位)
1	1	1	65536 (16 位)

注意事项 **1.** 设定采样时钟（转换时间）要满足表 **13-1** 所示的条件。选择转换时间时，需要考虑时钟频率误差。

2. 在 **16** 位ΔΣ 型 **A/D** 转换操作期间，禁止修改 **ADDCTL1** 寄存器。请确保在 **16** 位ΔΣ型 **A/D** 转换操作停止（**ADDCE = 0**）后写入。

3. 当 f_{SUB} 被选择作为 **16** 位ΔΣ型 **A/D** 采样时钟（f_{VP}）时，禁止设定并行模式。

转换时间由采样时钟 （f_{VP}）和采样计数（N）得出，所需计算如下：

采样时间 = $1/f_{VP} \times N$
初始化时间 = 1/操作时钟 + $1/f_{VP} \times 256$

操作时钟
ADDFS1-0 被选为 1，1： f_{SUB}
ADDFS1-0 被选为其他： f_{PRS}

在串行模式下
<第一次转换>
转换时间　= 初始化时间+ 采样时间
　= (1/操作时钟 + $1/f_{VP} \times 256$) + ($1/f_{VP} \times N$)

<第二次转换后>
转换时间　= 采样时间
　= $1/f_{VP} \times N$

在并行模式下
<第一次转换>
转换时间　= 初始化时间+ 采样时间
　= (1/操作时钟 + $1/f_{VP} \times 256$) + ($1/f_{VP} \times N$)

<第二次转换后>
转换时间　= 采样时间/4
　= $1/f_{VP} \times N/4$

f_{VP}： 采样时钟，N：16 位ΔΣ型 A/D 采样计数

注意事项　如果 **ADDCTL0** 被重写（包括相同的值），转换被认为是已经从该点重新开始，并且转换时间和第一次转换相同。

表 13-1. 采样时钟（采样时间）设定条件

ADDN2	AVREF条件	采样时钟 f_{VP}（16位分辨率的转换时间）
差分输入	3.5 V ≤ AVREF ≤ 5.5 V	1.25 MHz 最大值（52.42 ms 最小值）
	2.7 V ≤ AVREF < 3.5 V	625 kHz 最大值（104.85 ms 最小值）
单信号输入	2.85 V ≤ AVREF ≤ 5.5 V	625 kHz 最大值（104.85 ms 最小值）
	2.7 V ≤ AVREF < 2.85 V	525 kHz 最大值（124.83 ms 最小值）

表 13-2. 在设定条件下的转换时间示例

16位ΔΣ型A/D 采样次数的数量：N（分辨率）									
f_{PRS}	f_{VP}	65536 (16位)	32768 (15位)	16384 (14位)	8192 (13位)	4096 (12位)	2048 (11位)	1024 (10位)	256 (8位)
10 MHz	f_{PRS}/4	禁止设置	禁止设置	禁止设置	禁止设置	禁止设置	禁止设置	禁止设置	禁止设置
	f_{PRS}/8注 1	52.42 ms	26.21 ms	13.10 ms	6.55 ms	3.27 ms	1.63 ms	0.81 ms	0.20 ms
	f_{PRS}/16注 2	104.85 ms	52.42 ms	26.21 ms	13.10 ms	6.55 ms	3.27 ms	1.63 ms	0.41 ms
8 MHz	f_{PRS}/4	禁止设置	禁止设置	禁止设置	禁止设置	禁止设置	禁止设置	禁止设置	禁止设置
	f_{PRS}/8注 1	65.53 ms	32.76 ms	16.38 ms	8.19 ms	4.09 ms	2.04 ms	1.02 ms	0.25 ms
	f_{PRS}/16	131.07 ms	65.53 ms	32.76 ms	16.38 ms	8.19 ms	4.09 ms	2.04 ms	0.51 ms
5 MHz	f_{PRS}/4注 1	52.42 ms	26.21 ms	13.10 ms	6.55 ms	3.27 ms	1.63 ms	0.81 ms	0.40 ms
	f_{PRS}/8注 2	104.85 ms	52.42 ms	26.21 ms	13.10 ms	6.55 ms	3.27 ms	1.63 ms	0.81 ms
	f_{PRS}/16	209.71 ms	104.85 ms	52.42 ms	26.21 ms	13.10 ms	6.55 ms	3.27 ms	1.63 ms
4 MHz	f_{PRS}/4注 1	65.53 ms	32.76 ms	16.38 ms	8.19 ms	4.09 ms	2.04 ms	1.02 ms	0.25 ms
	f_{PRS}/8	131.07 ms	65.53 ms	32.76 ms	16.38 ms	8.19 ms	4.09 ms	2.04 ms	0.51 ms
	f_{PRS}/16	262.14 ms	131.07 ms	65.53 ms	32.76 ms	16.38 ms	8.19 ms	4.09 ms	1.02 ms
2 MHz	f_{PRS}/4	131.07 ms	65.53 ms	32.76 ms	16.38 ms	8.19 ms	4.09 ms	2.04 ms	0.51 ms
	f_{PRS}/8	262.14 ms	131.07 ms	65.53 ms	32.76 ms	16.38 ms	8.19 ms	4.09 ms	1.02 ms
	f_{PRS}/16	524.28 ms	262.14 ms	131.07 ms	65.53 ms	32.76 ms	16.38 ms	8.19 ms	2.04 ms
–	f_{SUB}/2	4 s	2 s	1 s	500 ms	250 ms	125 ms	62.5 ms	15.62 ms

注 1. 禁止设置差分输入模式 (2.7 V ≤ AVREF < 3.5 V) 和单信号输入模式，因为在该模式下不满足转换时间条件。

2. 禁止设置单信号输入模式（2.7 V ≤ AVREF < 2.85 V），因为在该模式下不满足转换时间条件。

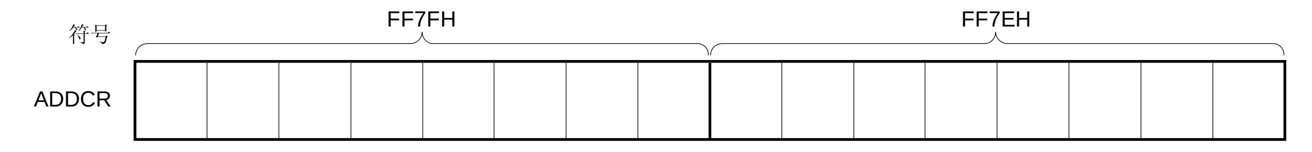
（3）　16 位 ΔΣ 型 A/D 转换结果寄存器（ADDCR）

该寄存器是 16 位寄存器，用于保存 A/D 转换的结果。每当 A/D 转换完成时，转换结果从ΔΣ A/D 电路装载到该寄存器。转换结果的高 8 位保存在 FF7FH 中，低 8 位保存在 FF7EH 中。

可以通过 16 位存储器操作指令来读取 ADDCR。

复位信号产生会将该寄存器清除为 0000H。

图 13-4. 16 位 ΔΣ型 A/D 转换结果寄存器（ADDCR）的格式



注意事项 **1.** 当设置 **N** 位分辨率后，转换结果将从高位开始保存，剩余的 **16 - N** 位固定为 **0**。

2. 如果转换完成中断和转换结果读取操作发生冲突，则转换的结果可能是不确定的。应该在转换完成中断产生之后，但在下一次转换完成之前，读取转换结果。

（4）　8 位 ΔΣ 型 A/D 转换结果寄存器 （ADDCRH）

该寄存器是 8 位寄存器，用于保存 A/D 转换结果。保存 16 位分辨率的高 8 位。

可以通过 8 位存储器操作指令来读取 ADDCRH。

复位信号产生会将该寄存器清除为 00H。

图 13-5. 8 位 ΔΣ 型 A/D 转换结果寄存器（ADDCRH）的格式

地址：　FF7FH　复位后：　00H　R

符号	7	6	5	4	3	2	1	0
ADDCRH								

注意事项　**如果转换完成中断和转换结果读取操作发生冲突，则转换的结果可能是不确定的。应该在转换完成中断产生之后，但在下一次转换完成之前，读取转换结果。**

（5）　16 位 ΔΣ 型 A/D 转换状态寄存器 （ADDSTR）

该寄存器用于保持已经完成 A/D 转换的通道。也可以用于检查是哪个通道已经完成转换。

可以通过 8 位存储器操作指令来读取 ADDSTR。

复位信号产生会将该寄存器清除为 00H。

图 13-6. 16 位 ΔΣ 型 A/D 转换状态寄存器 （ADDSTR）的格式

地址： FF75H　复位后： 00H　R

符号	7	6	5	4	3	2	1	0
ADDSTR	0	0	0	0	0	0	ADDIT1	ADDIT0

ADDIT1	ADDIT0	由 16 位ΔΣ型 A/D 转换器进行转换的通道	
		选择差分输入时	选择单信号输入时
0	0	DS0+/DS0-	DS0+
0	1	DS1+/DS1-	DS1+
1	0	DS2+/DS2-	DS2+

（6） A/D 端口配置寄存器 0（ADPC0）

该寄存器将 ANI0/P20/DS0- 至 ANI7/P27/REF+ 引脚切换为模拟输入（16 位 ΔΣ 型 A/D 转换器的模拟输入或 10 位逐次逼近型 A/D 转换器的模拟输入）或端口的数字 I/O。

可以通过 1 位或是 8 位存储器操作指令来设置 ADPC0 。

复位信号产生会将该寄存器设置为 08H。

图 13-7. A/D 端口配置寄存器 0（ADPC0）的格式

地址: FF8FH 复位后: 08H R/W

符号	7	6	5	4	3	2	1	0
ADPC0	0	0	0	0	ADPC03	ADPC02	ADPC01	ADPC00

ADPC03	ADPC02	ADPC01	ADPC00	数字 I/O (D)/模拟输入 (A: 逐次逼近，Δ: ΔΣ) 切换							
				P27/ ANI7/ REF+	P26/ ANI6/ REF-	P25/ ANI5/ DS2+	P24/ ANI4/ DS2-	P23/ ANI3/ DS1+	P22/ ANI2/ DS1-	P21/ ANI1/ DS0+	P20/ ANI0/ DS0-
0	0	0	0	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ
0	0	0	1	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A	D
0	0	1	0	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	A/Δ	D	D
0	0	1	1	A/Δ	A/Δ	A/Δ	A/Δ	A	D	D	D
0	1	0	0	A/Δ	A/Δ	A/Δ	A/Δ	D	D	D	D
0	1	0	1	A	A	A	D	D	D	D	D
0	1	1	0	A	A	D	D	D	D	D	D
0	1	1	1	A	D	D	D	D	D	D	D
1	0	0	0	D	D	D	D	D	D	D	D
其它情况				禁止设置							

注意事项 **1.** 使用端口模式寄存器 2（PM2），将用作 A/D 转换的通道设置为输入模式。

2. 如果一个引脚已经被 ADPC0 设置，不能再通过 ADS、ADDS1 或 ADDS0 将该引脚设置为数字 I/O。

3. 如果有数据写入 ADPC0，会产生一个等待周期。当 CPU 运行于副系统时钟，且外设硬件时钟处于停止状态时，不要对 ADPC0 写入数据。详情参见第三十三章 等待注意事项。

(7) 16 位 ΔΣ型 A/D 采样延迟时间设置使能寄存器

该寄存器用于使能对采样延迟时间的设置。

该寄存器的地址是直接指定的，且通过 8 位存储器操作指令来设置。

复位信号产生会将该寄存器清除为 00H。

备注　如果 A/D 转换精度足够，则该寄存器无需设置。

图 13-8. 16 位 ΔΣ型 A/D 采样延迟时间设置使能寄存器的格式

地址： FA26H　复位后： 00H　R/W

符号	7	6	5	4	3	2	1	0
	ADD0TEN	0	0	0	0	0	0	0

ADD0TEN	设置延迟时间的控制
0	禁止延迟时间设置
1	使能延迟时间设置

注意事项　**请确保在关闭 16 位ΔΣ型 A/D 电路的电源 (ADDPON = 0)并停止转换操作(ADDCE = 0)之后，设置该寄存器。**

(8) 16 位 ΔΣ型 A/D 采样延迟时间设置寄存器

该寄存器用于设置采样延迟时间。

通过设置优化的延迟时间，精度可以改善。

该寄存器的地址是直接指定的，且通过 8 位存储器操作指令来设置。

复位信号产生会将该寄存器清除为 20H。

备注　如果 A/D 转换精度足够，则该寄存器无需设置。

图 13-9. 16 位 ΔΣ型 A/D 采样延迟时间设置寄存器的格式

地址： FA27H　复位后： 20H　R/W

符号	7	6	5	4	3	2	1	0
	0	ADD0DLY2	ADD0DLY1	ADD0DLY0	0	0	0	0

ADD0DLY2	ADD0DLY1	ADD0DLY0	设置延迟时间 [nsec]
0	0	0	2
0	0	1	4
0	1	0	6 (默认)
0	1	1	8
1	0	0	10
1	0	1	12
1	1	0	14
1	1	1	16

注意事项　**请确保在关闭 16 位ΔΣ型 A/D 电路的电源 (ADDPON = 0)并停止转换操作(ADDCE = 0)之后，设置该寄存器，并使能延迟时间设置(ADD0TEN = 1)。**

（9） 端口模式寄存器 2 （PM2）

使用 ANI0/P20/DS0–至 ANI7/P27/REF+ 引脚作为模拟输入端口时，将 PM20 至 PM27 设置为 1。此时，P20 至 P27 的输出锁存器的内容可能为 0 或 1。

如果 PM20 至 PM27 被设置为 0，则它们不能被用作模拟输入端口引脚。

可以通过 1 位或 8 位存储器操作指令来设置 PM2。

复位信号产生会将该寄存器清除为 FFH。

图 13-10. 端口模式寄存器 2（PM2）的格式

地址： FF22H 复位后： FFH R/W

符号	7	6	5	4	3	2	1	0
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20

PM2n	P2n 引脚输入/输出模式选择（n = 0 至 7）
0	输出模式（输出缓冲期打开）
1	输入模式（输出缓冲期关闭）

根据 ADPC0、PM2、ADS 和 ADDCTL0 的设置如，P20/ANI0/DS0– 至 P27/ANI7/REF+ 引脚如下。

表 13-3. P20/ANI0/DS0– 至 P27/ANI7/REF+ 引脚的功能设置

ADPC0	PM2	ADS	ADDCTL0	P20/ANI0/DS0–至 P27/ANI7/REF+ 引脚
模拟输入选择	输入模式	不选择 ANI	不选择 DSn±.	模拟输入（不进行转换）
		选择 ANI	不选择 DSn±.	模拟输入（使用逐次逼近型 A/D 转换器进行转换）
		不选择 ANI	选择 DSn±.	模拟输入（使用ΔΣ 型 A/D转换器进行转换）
		选择 ANI	选择 DSn±.	禁止设定
	输出模式	–		禁止设定
数字 I/O 选择	输入模式	–		数字输入
	输出模式	–		数字输出

备注 n = 0 至 2

13.4 16 位 ΔΣ型A/D转换器的电路配置示例

图 13-11 显示使用 16 位 ΔΣ型 A/D 转换器时的电路配置示例。

图 13-11. 使用 16 位 ΔΣ型 A/D 转换器（差分输入）时的电路配置示例

13.5 16 位 ΔΣ型 A/D转换器操作

13.5.1 16 位 ΔΣ型 A/D 转换器基本操作

<1>　设置 ADD0TEN[注1] = 1 （使能延迟时间设置）[注2]
<2>　使用 ADD0DLY2 至 ADD0DLY0[注1] 来设置延迟时间[注2]
<3>　设置 ΔΣ A/D 转换目标通道
<4>　设置 ADDPON 为 1 （ΔΣ A/D 电源打开）。
<5>　通过 ADDCTL1 和 ADDCTL0 寄存器设置转换操作模式，比如输入模式，操作模式和采样计数。
<6>　当 ADDCE 被设置为 1 时，转换操作开始，但是该操作必须在 ADDPON 被设置为 1 后经过至少 1.2μs 才能进行。（在 ADDPON 被置 1 后经过至少 1.2μs 之前，如果将 ADDCE 置 1，转换操作也会开始，但是这种情况下无法保证转换结果）。
<7>　当转换完成时，会产生一个中断（INTDSAD），结果被保存到 ADDCR 寄存器。读取 ADDCR 寄存器的值。
<8>　如果 ADDCE 没有被设置为 0（转换操作停止），重复步骤 5。如果要停止转换操作，设置 ADDCE 为 0。
<9>　不使用ΔΣ A/D 时，为了降低电流，设置 ADDPON 为 0（ΔΣ A/D 电源关闭）。

注　**1.** 直接指定地址，并使用 8 位存储器操作指令写入 ADD0TEN 和 ADD0DLY2 至 ADD0DLY0。
2. 如果 A/D 转换精度足够，则该寄存器无需设置。
3. 在转换操作期间，禁止对 ADDCTL1 进行写操作。
如果在转换期间，正在进行ΔΣ A/D 转换的引脚设置被改变，那么转换结果不保存。如果目标引脚的设置被改变，则需要重新启动转换操作。

13.5.2 16 位 ΔΣ型 A/D 转换器的操作模式

16 位ΔΣ型 A/D 转换器可以设定以下几种操作模式。

（1）差分输入模式/单信号输入模式
16 位 ΔΣ型 A/D 转换器的输入模式可以设定为差分输入模式或单信号输入模式，差分输入模式的精度要高于单信号输入模式。当使用差分输入模式时，将模拟信号输入到 DSn-和 DSn+引脚。当使用单信号输入模式时，将模拟信号输入到 DSn+引脚，并设置 DSn- 引脚的电位与 V_{SS} 和 AV_{SS} 相同。在差分输入模式下，确保 DSn- 和 DSn+输入电压的中间值是 0.5 REF+。

（2）16 位 ΔΣ 型 A/D 高精度模式
可以选择高精度模式 ON 或 OFF 作为 16 位 ΔΣ型 A/D 转换器的转换精度模式，当设置高精度模式 ON 时，精度高于设置高精度模式 OFF 时。

表 13-4. 输入电压范围

		输入到DSn+的电压范围	输入到DSn-的电压范围
差分输入	高精度模式ON	0.5 ×（REF+）+ X1	0.5 ×（REF+）– X1
	高精度模式OFF	0.5 ×（REF+）+ X2	0.5 ×（REF+）– X2
信号输入	高精度模式ON	0.1 ×（REF+）至 0.9 ×（REF+）	固定为 AV_{SS}
	高精度模式OFF	0 至 REF+	

备注　X1 = –0.4 ×（REF+）至 0.4 ×（REF+）
X2 = –0.5 ×（REF+）至 0.5 ×（REF+）
n = 0 至 2

图 13-12. 应用电路的示例

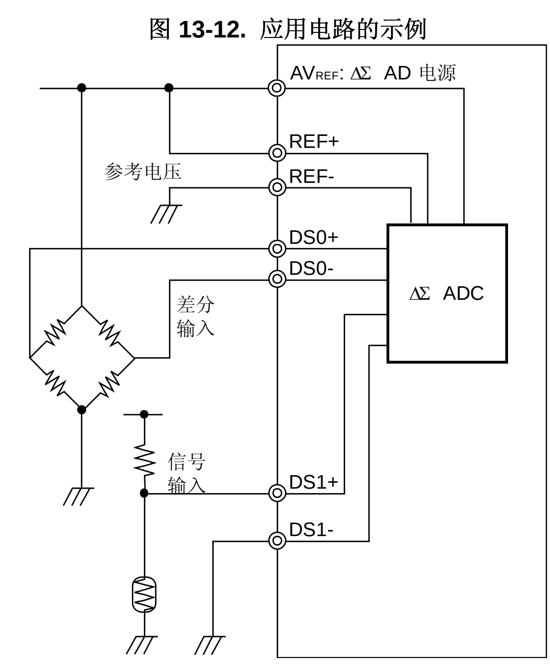
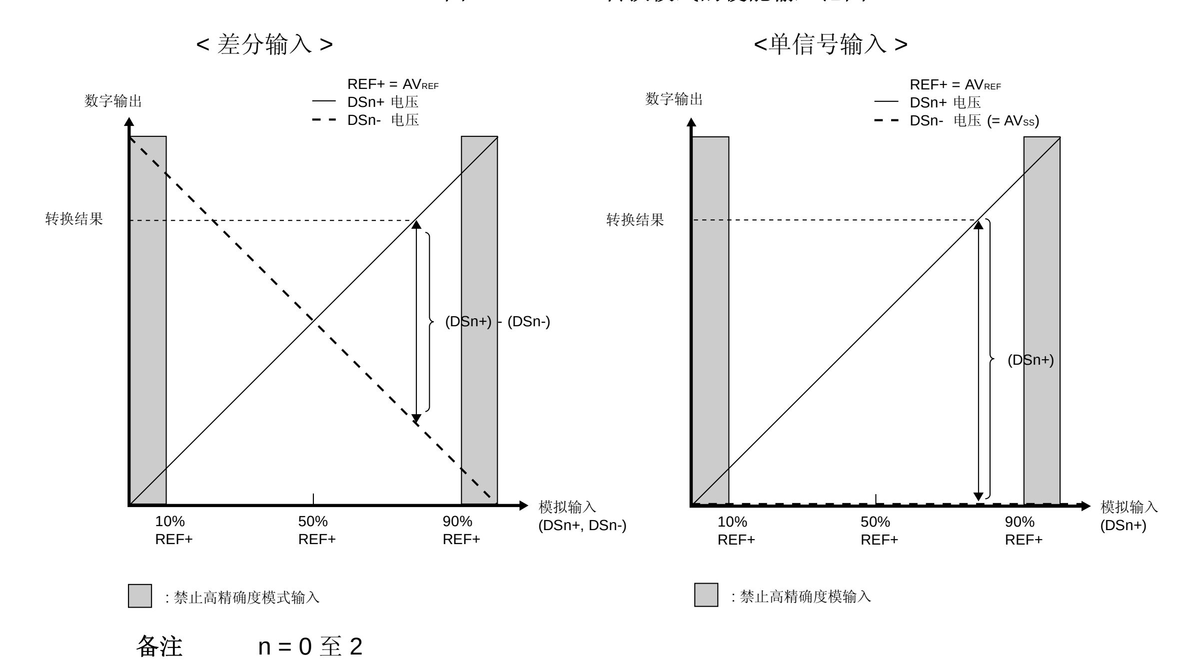


图 13-13. A/D 转换模式的使能输入范围

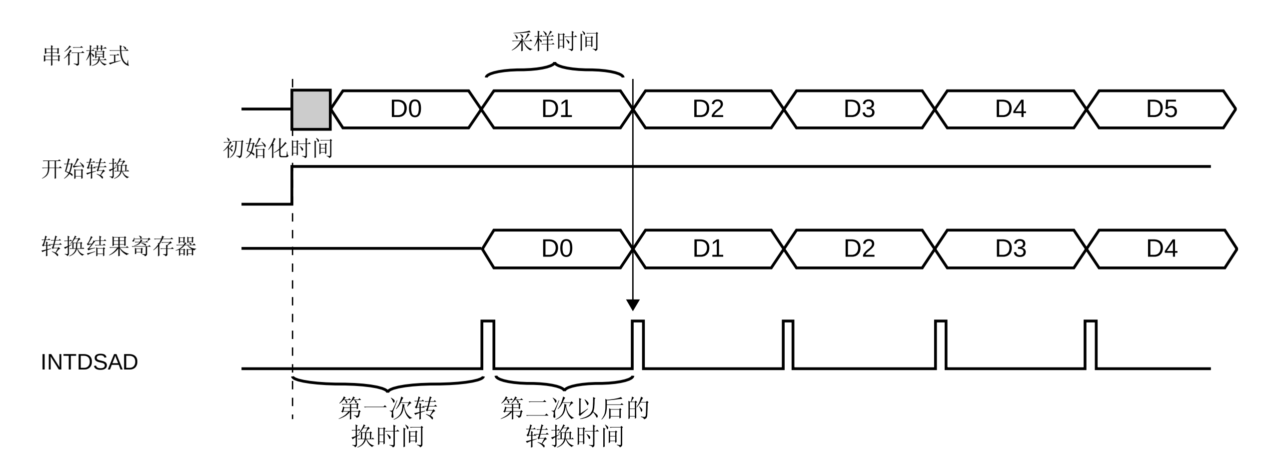


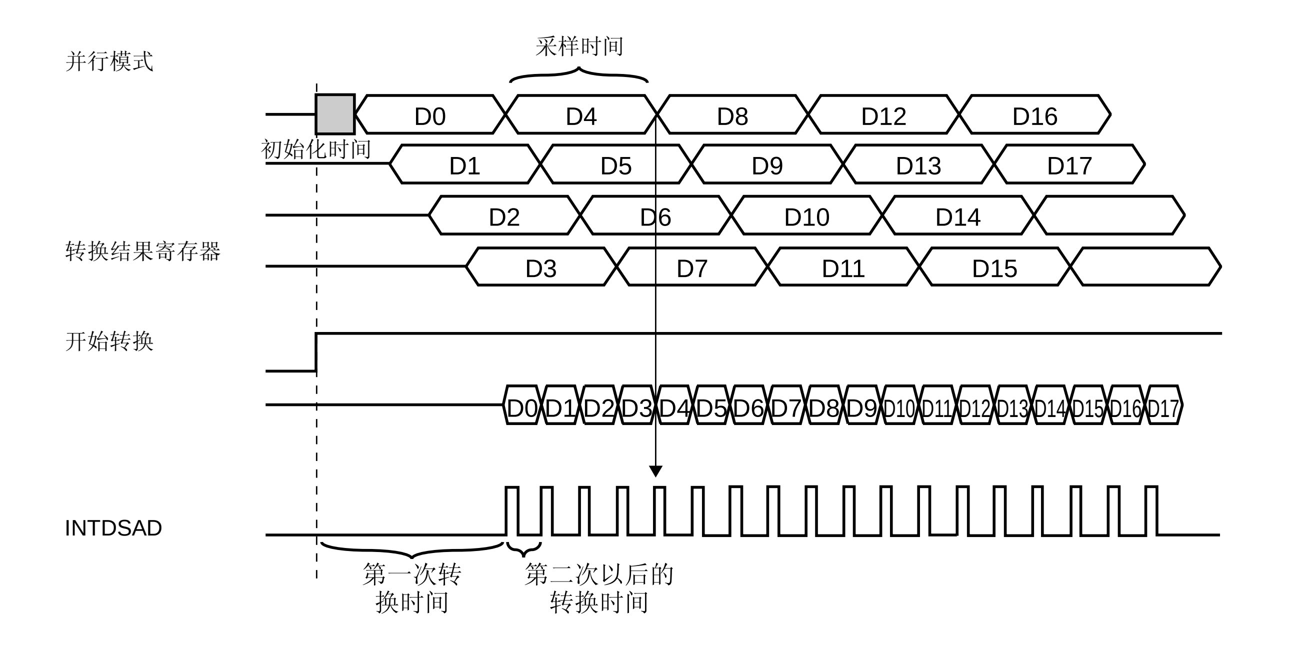
备注　n = 0 至 2

（3）串行模式/并行模式

16 位ΔΣ型 A/D 转换器的输入模式可以选择串行或是并行模式。并行模式可以将转换时间减少为串行模式的四分之一。但是并行模式第一次转换的时间与串行模式第一次转换的时间相同，而且，采样时间本身也和串行模式的采样时间相同。

图 13-14. 转换时间和采样时间





13.6 ΔΣ型 A/D转换器特征表的阅读方法

这里解释 ΔΣ型 A/D 转换器特有的专用术语。

（1） 分辨率

这是可以识别的最小的模拟输入电压，即数字输出的每位占模拟输入电压的百分比，称为 1LSB（最低有效位）。1 LSB 对满量程的比率被表示为%FSR（满标度量程）。

当分辨率为 16 位时，1LSB 表示如下。

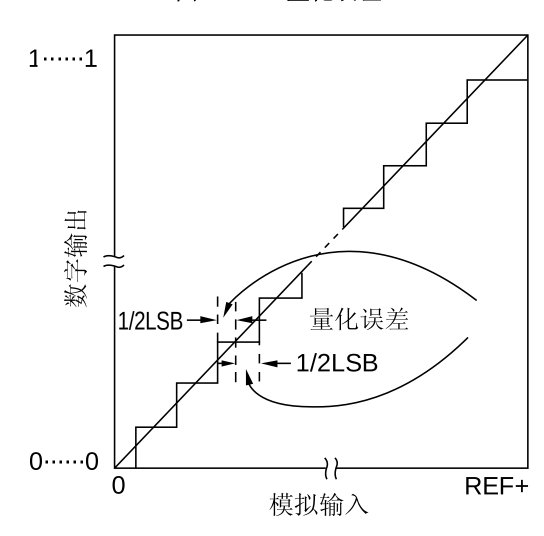
$1LSB = 1/2^{16} = 1/65536$
$\cong 0.0015\%FSR$

（2） 量化误差

当模拟值转换为数字值时，会不可避免的产生 ± 1/2 LSB 的误差。在一个 A/D 转换器中，相差 ±1/2LSB 的模拟输入电压被转换为相同的数字值，所以量化误差不可避免。

注意量化误差不属于特征表中的偏移误差、增益误差、积分线性误差和微分线性误差。

图 13-15. 量化误差

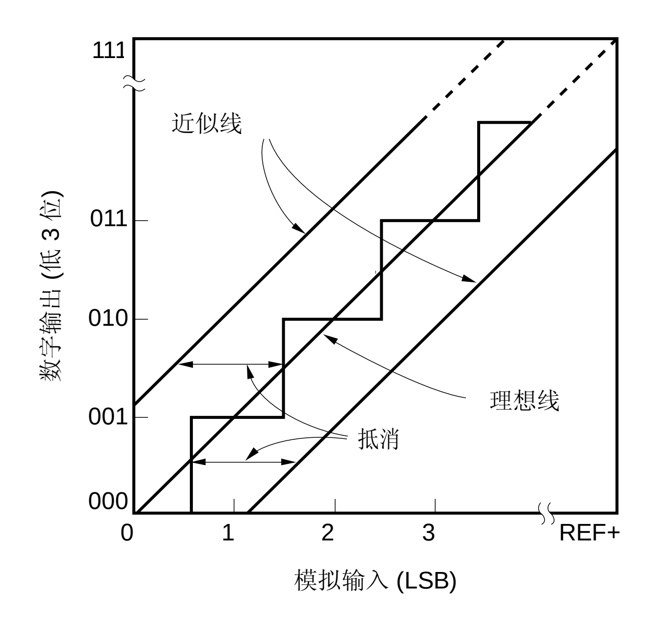


（3） 偏移误差（单信号输入）

偏移误差表示实际测量的模拟输入电压值与理论值（1/2LSB）之间的差别。

如果实际测量值大于理论值，偏移误差表示实际测量的模拟输入电压值的近似线与理论值（3/2LSB）之间的差别。

图 13-16. 偏移误差（单信号输入）



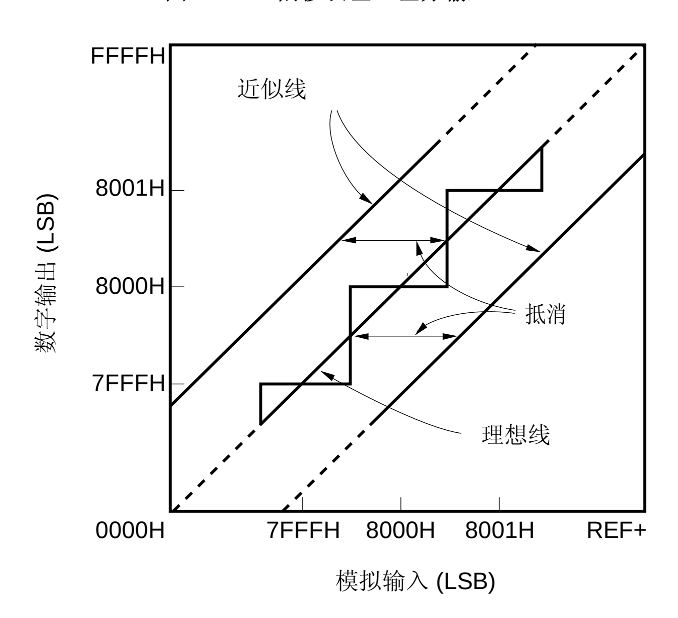
（4） 偏移误差（差分输入）

偏移误差表示实际测量的模拟输入电压值与理论值（1/2 满量程）之间的差别。

在 16 位分辨率的情况下，偏移误差表示实际测量的模拟输入电压值的近似线与理论值（8000H – 1/2 LSB）之间的差别。

如果近似线大于理论值，偏移误差表示实际测量的模拟输入电压值的近似线与理论值（8000H + 1/2 LSB）之间的差别。

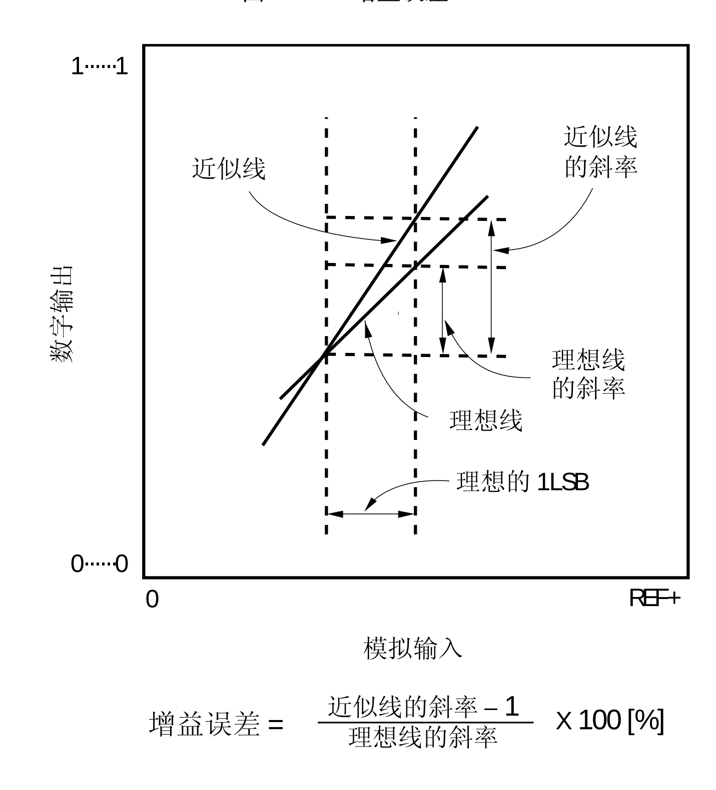
图 13-17. 偏移误差（差分输入）



（**5**） 增益误差

增益误差表示理想斜率和近似线斜率的比率。

图 13-18. 增益误差

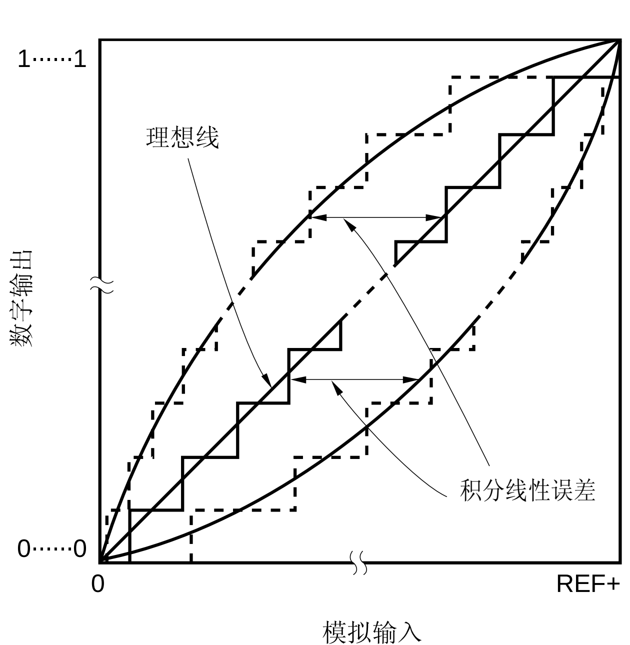


$$增益误差 = \frac{近似线的斜率 - 1}{理想线的斜率} \times 100\ [\%]$$

（**6**） 积分线性误差

积分线性误差表示转换特征偏离近似线的程度。它表示当偏移误差和增益误差均为 0 时，实际测量值与理想直线之间误差的最大值。

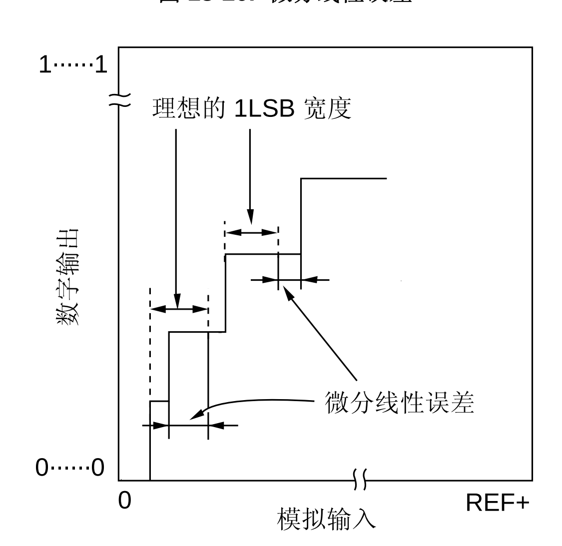
图 13-19. 积分线性误差



（7） 微分线性误差

当代码输出的理想宽度为 1LSB 时，微分线性误差表示实际测量值与理想值之间的差别。但是，不包括增益误差。

图 13-20. 微分线性误差



（8） 转换时间

转换时间表示从开始转换或获取转换结果到获取下一次转换结果所经历的时间。

详情参见**图 13-14. 转换时间和采样时间。**

（9） 采样时间

采样时间是执行一次转换所需的时间。

详情参见**图 13-14. 转换时间和采样时间。**

（10） 近似线

近似线是对实际测量值应用最小二乘方定义的线。

13.7 16 位 ΔΣ型 A/D 转换器的注意事项

（1） 设置待机模式

设置为 STOP 模式之前，将 16 位 ΔΣ型 A/D 转换器控制寄存器 0 （ADDCTL0） 的第 7 位（ADDPON）和第 6 位（ADDCE）清除为 0。

当从待机状态重启，将中断请求标志寄存器 1L（IF1L）的第 6 位（DSADIF）清除为 0，并开始操作。

（2） DS0–，DS0+，DS1–，DS1+，DS2–和 DS2+的输入范围

注意 DS0–，DS0+，DS1–，DS1+，DS2–和 DS2+输入电压的额定范围。如果输入到模拟输入通道的电压大于等于 REF+ （AV_{REF}），或者小于等于 REF– （AV_{SS}）（即使在绝对最大额定范围之内），则该通道的转换值不确定。此外，其他通道的转换值也可能会受影响。

（3） 冲突操作

<1> 在转换结束时，对 A/D 转换结果寄存器（ADDCR，ADDCRH）的写入操作和通过指令读取 ADDCR 或 ADDCRH 的操作之间的冲突。
ADDCR 或 ADDCRH 读操作优先。在执行读操作之后，新的转换结果被写入 ADDCR 或 ADDCRH。

<2> 在转换结束时，对 ADDCR 或 ADDCRH 的写入操作和对 16 位 ΔΣ型 A/D 转换器控制寄存器 0（ADDCTL0）的写入操作或对 A/D 端口配置寄存器 0（ADPC0）的写入操作之间的冲突。
ADDCTL0 或 ADPC0 的写入操作优先。不执行对 ADDCR 或 ADDCRH 的写入，也不会产生转换结束中断信号（INTDSAD）。

（4） 噪声处理

为了保持规范的精度，必须要注意输入到 DS0–，DS0+，DS1–，DS1+，DS2–，DS2+，REF– （AV_{SS}）和 REF+（AV_{REF}）引脚的噪声。

<1> 在电源上连接一个具有较低等效电阻和优良频率响应的电容。
<2> 模拟输入源的输出阻抗越大，影响就越大。为了降低噪声，推荐连接外部电容。
<3> 在转换期间不要切换引脚。
<4> 如果在转换启动后立即设置 HALT 模式，则可能会改善精度。

（5） DS0–/ANI0/P20，DS0+/ANI1/P21，DS1–/ANI2/P22，DS1+/ANI3/P23，DS2–/ANI4/P24，DS2+/ANI5/P25，REF–/ANI6/P26 和 REF+/ANI7/P27

<1> 模拟输入引脚（DS0–，DS0+，DS1–，DS1+，DS2–，DS2+，REF–和 REF+）也可以用作输入端口引脚（P20 至 P27）。
当通过 DS0–/DS0+，DS1–/DS1+或 DS2–/DS2+中的任意一组选中的通道执行 16 位 ΔΣ型 A/D 转换时，转换过程进行中不要访问 P20 至 P27；否则转换分辨率可能会降低。推荐从离 REF+ 最远的 DS0–/ANI0/P20 开始，选择 P20 至 P27 的引脚作为数字 I/O 端口。

<2> 如果在 16 位 ΔΣ型 A/D 转换期间，P20 至 P27 中的任意引脚被用作数字输入/输出，则由于噪声耦合，有可能得不到预期的 A/D 转换值。因此，在进行 A/D 转换期间，请确保不要从引脚 P20 至 P27 输入或输出数字脉冲。

（6）　**DS0+，DS1+，DS2+，DS0–，DS1–和 DS2– 引脚的输入阻抗**

采样期间 A/D 转换器对采样电容充电，以便采样。

因此，不进行采样时仅有漏电流流过，而在采样期间则有电容充电的电流流过，因此，输入阻抗的波动会根据是否在进行采样或位于其他状态。

为了确保采样有效，推荐保持模拟输入源的输出阻抗小于等于 5 kΩ。

（7）　**AVREF 引脚的输入阻抗**

当 16 位 ΔΣ型 A/D 转换操作时，有电流流入 AVREF 引脚。

如果参考电压源的输出阻抗很大，AVREF pin/REF+ 引脚和 AVSS pin/REF– 引脚之间的参考电压误差增大。

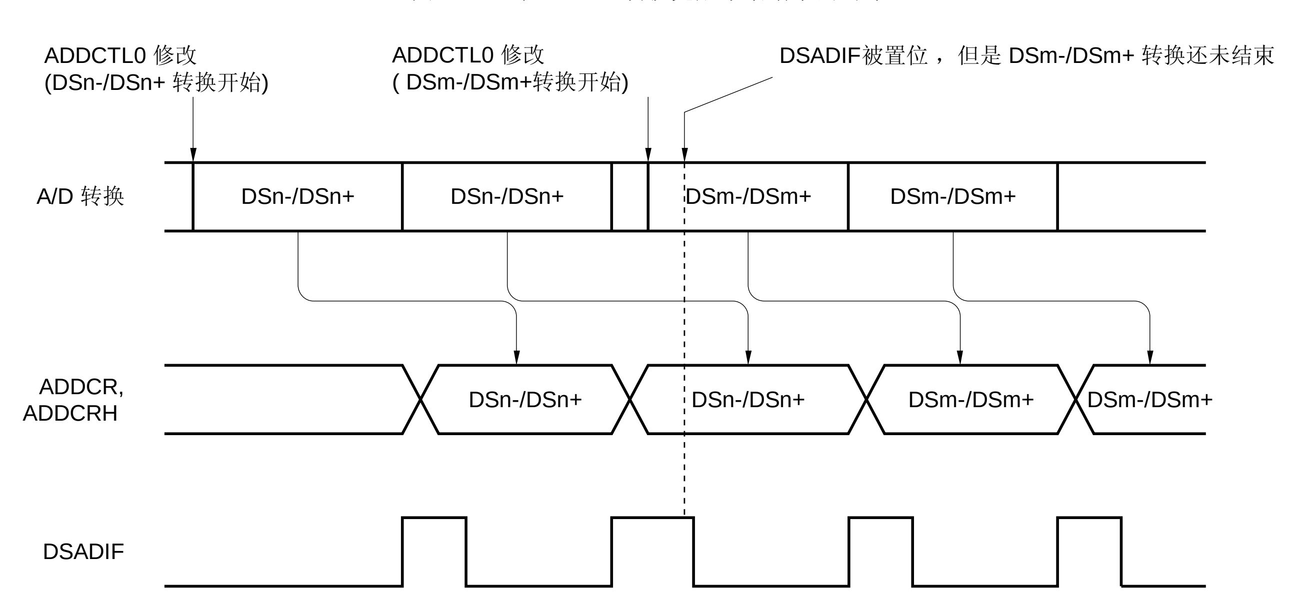
（8）　**中断请求标志 （DSADIF）**

即使 16 位 ΔΣ型 A/D 转换器控制寄存器 0（ADDCTL0）的第 1 位和第 0 位被改变，中断请求标志（DSADIF）也不会被清除。

因此，如果在 A/D 转换期间有一个模拟输入引脚被改变，则就在 ADDCTL0 被修改之前，改变前的模拟通道的 A/D 转换结果和 ADIF 可能刚被设置。因此需要特别注意，此时，当 ADDCTL0 重写后立即读取 ADIF 时，即使修改后的模拟输入的实际 A/D 转换尚未结束，也会设置 DSADIF。

当 A/D 转换停止后又重新恢复时，在 A/D 转换操作被恢复之前清除 DSADIF。

图 13-21. 产生 A/D 转换完成中断请求的时序



备注　**1.** n = 0 至 2
　　　2. m = 0 至 2

（9）　A/D 转换刚启动后的转换结果

若在 ADDPON 置 1 后的 1.2 μs 内对 ADDCE 置 1，或在 ADDPON 位= 0 时对 ADDCE 置 1，那么 A/D 转换刚启动后第一次 A/D 转换值可能不在额定范围内。可以采取措施，比如轮询 A/D 转换结束中断请求（INTDSAD），并删除第 1 次转换结果。

（10）　内部等效电路

模拟输入模块的等效电路如下所示。

图 13-22. DSn– 和 DSn+ 引脚的内部等效电路

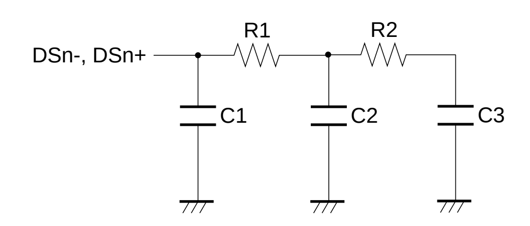


表 13-5. 内部等效电路的电阻和电容值（参考值）

AV_{REF}	R1	R2	C1	C2	C3
4.0 V ≤ AV_{REF} ≤ 5.5 V	8.1 kΩ	6.8 kΩ	8 pF	1.3 pF	0.22 pF
2.7 V ≤ AV_{REF} < 4.0 V	31 kΩ	36 kΩ	8 pF	1.3 pF	0.22 pF

备注　**1.** 表 13-5 列出的电阻和电容值不是保证值。
　　　2. n = 0 至 2

（11）　同时使用 10 位逐次逼近型 A/D 转换器和 16 位ΔΣ型 A/D 转换器

同时使用 10 位逐次逼近型 A/D 转换器和 16 位ΔΣ型 A/D 转换器时，A/D 转换器的精度会降低。

当 10 位逐次逼近型 A/D 转换器操作期间，应该停止 16 位ΔΣ型 A/D 转换器，否则将无法保证操作精度。同样，当 16 位ΔΣ型 A/D 转换器操作期间，应该停止 10 位逐次逼近型 A/D 转换器。（不要同时操作）

14.1 串行接口UART0 的功能

串行接口 UART0 有以下两种模式。

（1）　操作停止模式

不执行串口通信时，使用该模式，可以降低功耗。

详情参见 **14.4.1 操作停止模式。**

（2）　异步串行接口（UART）模式

该模式的功能概括如下。

详情参见 **14.4.2 异步串行接口（UART）模式**和 **14.4.3 专用波特率发生器。**

- 最大传输率：　625 kbps
- 双引脚配置　TXD0：　发送数据输出引脚
RxD0：　接收数据输入引脚
- 通信数据的长度可以选择 7 位或 8 位。
- 专用片上 5 位波特率发生器，允许设置任意波特率。
- 发送和接收可以独立执行（全双工操作）。
- 固定为 LSB-先行通信

注意事项

1. **如果提供给串行接口 UART0 的时钟未停止（例如 HALT 模式下），则继续正常操作。如果提供的时钟已经停止（例如 STOP 模式下），则每个寄存器都停止操作，并且保持时钟停止前的瞬时值。TxD0 引脚也保持时钟停止前的瞬时值，并输出该电平。但是，在时钟供应恢复之后，该操作无法保证。因此，需要复位该电路，使得 POWER0 = 0、RXE0 = 0 且 TXE0 = 0。**
2. **设置 POWER0 = 1，然后设置 TXE0 = 1（发送）或 RXE0 = 1（接收），开始通信。**
3. **通过 BRGC0 设置的基准时钟（f_{XCLK0}）使 TXE0 与 RXE0 同步。要再次使能发送或接收，在 TXE0 或 RXE0 被清除为 0 后，至少经过两个基准时钟，再将 TXE0 或 RXE0 设置为 1。如果在两个基准时钟时以内对 TXE0 或 RXE0 置位，则发送电路或接收电路可能无法初始化。**
4. **设置 TXE0=1 之后，经过至少一个基准时钟，才能将发送数据设置到 TXS0。**

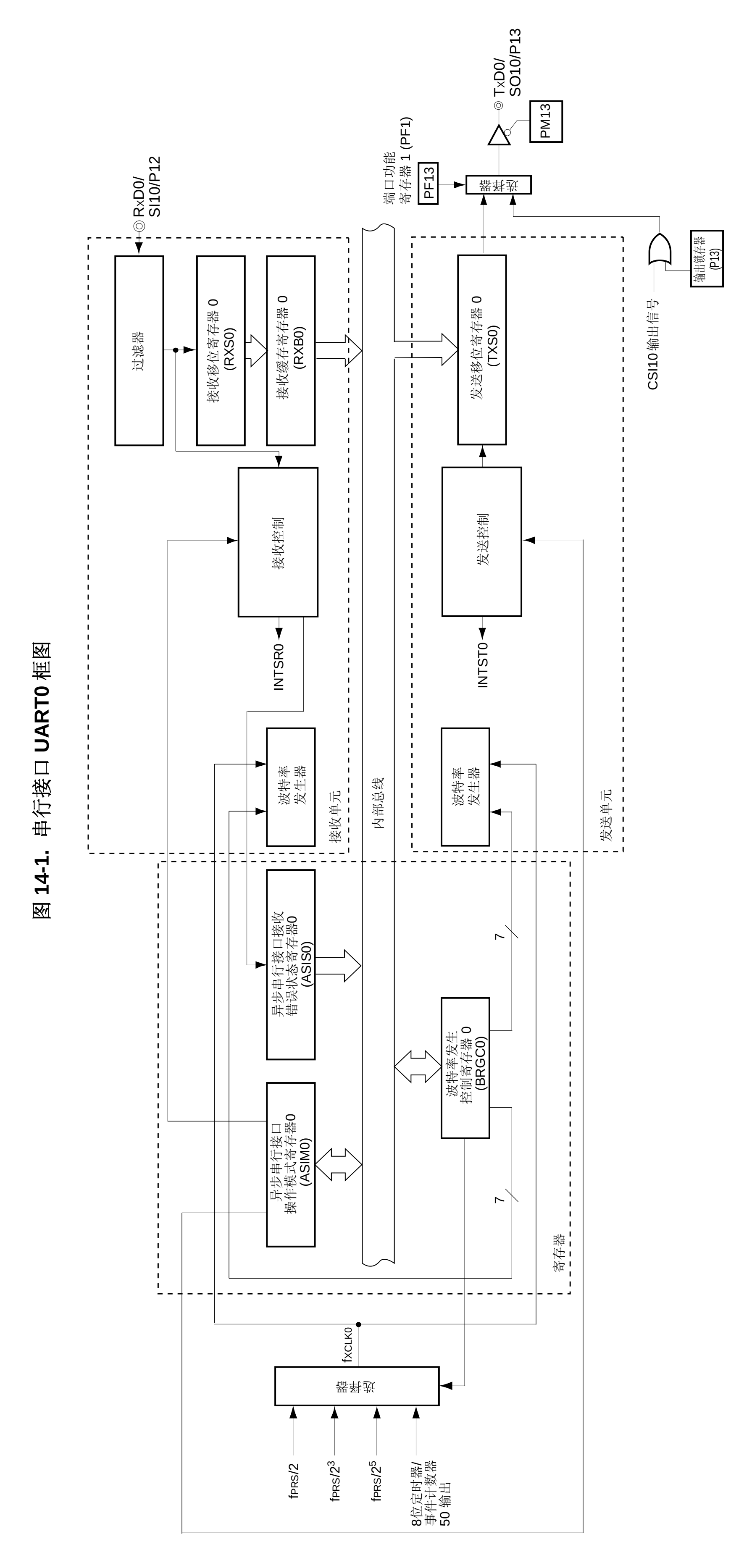
14.2 串行接口UART0 的配置

串行接口 UART0 包括以下硬件。

表 14-1. 串行接口 UART0 的配置

项目	配置
寄存器	接收缓冲寄存器 0（RXB0） 接收移位寄存器 0（RXS0） 发送移位寄存器 0（TXS0）
控制寄存器	异步串行接口操作模式寄存器 0（ASIM0） 异步串行接口接收错误状态寄存器 0（ASIS0） 波特率发生器控制寄存器 0（BRGC0） 端口功能寄存器 1（PF1） 端口模式寄存器 1（PM1） 端口寄存器 1（P1）

图 14-1. 串行接口 UART0 框图



（1） 接收缓冲寄存器 0（RXB0）

该 8 位寄存器用于存储由接收移位寄存器 0（RXS0）转换的并行数据。

每次完成 1 字节数据的接收时，新的接收数据就从接收移位寄存器 0（RXS0）传送到该寄存器。

如果数据长度被设定为 7 位，则将接收到的数据传送到 RXB0 的第 0 位至第 6 位，而 RXB0 的 MSB 位总是 0。

如果发生溢出错误（OVE0），则不把接收数据传送到 RXB0 中。

可以通过 8 位存储器操作指令来读取 RXB0。不能向该寄存器写入数据。

复位信号的产生和 POWER0 = 0 会将该寄存器设置为 FFH。

（2） 接收移位寄存器 0（RXS0）

该寄存器用于将输入到 RxD0 引脚的串行数据转换为并行数据。

程序不能直接操作 RXS0。

（3） 发送移位寄存器 0（TXS0）

该寄存器用于设置发送数据。当数据写入 TXS0 时，启动发送，串行数据从 TxD0 引脚发送。

可以通过 8 位存储器操作指令来写入 TXS0。不能读取该寄存器。

复位信号的产生、POWER0 = 0 或 TXE0 = 0 会将该寄存器设置为 FFH。

注意事项 **1. 设置 TXE0=1 之后，经过至少一个基准时钟（f_{XCLK0}），才能将发送数据设置到 TXS0。**

2. 在产生发送完成中断信号（INTST0）之前，不要将下一个发送数据写入 TXS0。

14.3 控制串行接口UART0 的寄存器

串行接口 UART0 由以下 6 个寄存器控制。

- 异步串行接口操作模式寄存器 0（ASIM0）
- 异步串行接口接收错误状态寄存器 0（ASIS0）
- 波特率发生器控制寄存器 0（BRGC0）
- 端口功能寄存器 1（PF1）
- 端口模式寄存器 1（PM1）
- 端口寄存器 1（P1）

（1） 异步串行接口操作模式寄存器 0（ASIM0）

该 8 位寄存器用于控制串行接口 UART0 的串行通信操作。

可以通过 1 位或 8 位存储器操作指令来设置该寄存器。

复位信号的产生会将该寄存器设置为 01H。

图 14-2. 异步串行接口操作模式寄存器 0（ASIM0）的格式（1/2）

地址： FF70H 复位后： 01H R/W

符号	<7>	<6>	<5>	4	3	2	1	0
ASIM0	POWER0	TXE0	RXE0	PS01	PS00	CL0	SL0	1

POWER0	内部操作时钟的使能/禁止操作
0[注1]	禁止内部操作时钟的操作（时钟固定为低电平）并异步复位内部电路[注2]
1	使能内部操作时钟的操作

TXE0	使能/禁止发送
0	禁止发送（同步复位发送电路）
1	使能发送

RXE0	使能/禁止接收
0	禁止接收（同步复位接收电路）
1	使能接收

注　**1.** 当 POWER0 = 0 时，从 RxD0 引脚的输入固定为高电平。

2. 异步串行接口接收错误状态寄存器 0（ASIS0）、发送移位寄存器 0（TXS0）和接收缓冲寄存器 0（RXB0）被复位。

图 **14-2.** 异步串行接口操作模式寄存器 **0**（**ASIM0**）的格式（**2/2**）

PS01	PS00	发送操作	接收操作
0	0	不输出校验位	无校验接收
0	1	输出零校验	零校验接收[注]
1	0	输出奇校验	按奇校验判断
1	1	输出偶校验	按偶校验判断

CL0	指定发送/接收数据的字符长度
0	数据的字符长度 = 7 位
1	数据的字符长度 = 8 位

SL0	指定发送数据停止位的数量
0	停止位数量 = 1
1	停止位数量 = 2

注　如果选择"按零校验接收"，则不判断校验。因此异步串行接口接收错误状态寄存器 0（ASIS0）的第 2 位（PE0）不被置位，这样就不会产生错误中断。

注意事项
1. 要启动发送，将 **POWER0** 设置为 **1**，然后将 **TXE0** 设置为 **1**。要停止发送，先将 **TXE0** 清除为 **0**，然后将 **POWER0** 清除为 **0**。
2. 要启动接收，将 **POWER0** 设置为 **1**，然后将 **RXE0** 设置为 **1**。要停止接收，先将 **RXE0** 清除为 **0**，然后将 **POWER0** 清除为 **0**。
3. 当 **RxD0** 引脚输入为高电平时，先将 **POWER0** 设置为 **1**，然后将 **RXE0** 设置为 **1**。当输入为低电平时设置 **POWER0** 为 **1**，且设置 **RXE0** 为 **1** 时，可以启动接收。
4. 通过 **BRGC0** 设置的基准时钟（f_{XCLK0}）使 **TXE0** 与 **RXE0** 同步。要再次使能发送或接收，在 **TXE0** 或 **RXE0** 被清除为 **0** 后，至少经过两个基准时钟，再将 **TXE0** 或 **RXE0** 设置为 **1**。如果在两个基准时钟时以内对 **TXE0** 或 **RXE0** 置位，则发送电路或接收电路可能无法初始化。
5. 设置 **TXE0=1** 之后，经过至少一个基准时钟，才能将发送数据设置到 **TXS0**。
6. 在重写 **PS01**、**PS00** 和 **CL0** 位之前，先将 **TXE0** 和 **RXE0** 位清除为 **0**。
7. 重写 **SL0** 位时，必须确保 **TXE0 = 0**。总是按照**"停止位的数量=1"**执行接收操作，因此，不会受到 **SL0** 设置值的影响。
8. 请确保设置第 **0** 位为 **1**。

（2） 异步串行接口接收错误状态寄存器 0 （ASIS0）

该寄存器用于表示串行接口 UART0 接收操作完成时的错误状态，包括 3 个错误标志位（PE0，FE0，OVE0）。

只能通过 8 位存储器操作指令来读取该寄存器。

复位信号的产生，或将 ASIM0 的第 5 位（RXE0）或第 7 位（POWER0）清除为 0，都会将该寄存器清除为 00H。读取该寄存器时，读到的值是 00H。如果产生一个接收错误，则先读取 ASIS0 然后再读取接收缓冲寄存器 0（RXB0），可以清除错误标志。

图 14-3. 异步串行接口接收错误状态寄存器 0（ASIS0）的格式

地址： FF73H 复位后： 00H R

符号	7	6	5	4	3	2	1	0
ASIS0	0	0	0	0	0	PE0	FE0	OVE0

PE0	表示校验错误的状态标志位
0	如果 POWER0 = 0 且 RXE0 = 0，或 ASIS0 寄存器被读取
1	如果发送数据的校验位和接收完成时的校验位不匹配

FE0	表示帧错误的状态标志位
0	如果 POWER0 = 0 且 RXE0 = 0，或 ASIS0 寄存器被读取
1	如果在接收完成时未检测到停止位

OVE0	表示溢出错误的状态标志位
0	如果 POWER0 = 0 且 RXE0 = 0，或 ASIS0 寄存器被读取
1	如果接收数据被设置在 RXB0 寄存器中，并且在读取该数据之前，下一个接收操作已完成

注意事项

1. **根据异步串行接口操作模式寄存器 0（ASIM0）的 PS01 和 PS00 位的设置值，PE0 位的操作也有所不同。**
2. **无论停止位的数量如何设置，只能检测接收数据的第 1 位作为停止位。**
3. **如果发生溢出错误，则下一个接收数据不被写入接收缓冲寄存器 0（RXB0），而是被丢弃。**
4. **如果从 ASIS0 读取数据，会产生一个等待周期。当 CPU 运行于副系统时钟操作且外设硬件时钟处于停止状态时，不要从 ASIS0 读取数据。详情参见第三十三章 等待注意事项。**

（3） 波特率发生器控制寄存器 0（BRGC0）

该寄存器用于选择串行接口 UART0 的基准时钟和 5 位计数器的分频值。

可以通过 1 位或 8 位存储器操作指令来设置 BRGC0。

复位信号的产生会将该寄存器设置为 1FH。

图 14-4. 波特率发生器控制寄存器 0（BRGC0）的格式

地址： FF71H 复位后： 1FH R/W

符号	7	6	5	4	3	2	1	0
BRGC0	TPS01	TPS00	0	MDL04	MDL03	MDL02	MDL01	MDL00

TPS01	TPS00	基准时钟 （f_{XCLK0}） 的选择[注 1]				
			f_{PRS} = 2 MHz	f_{PRS} = 5 MHz	f_{PRS} = 8 MHz	f_{PRS} = 10 MHz
0	0	TM50 输出[注 2]				
0	1	$f_{PRS}/2$	1 MHz	2.5 MHz	4 MHz	5 MHz
1	0	$f_{PRS}/2^3$	250 kHz	625 kHz	1 MHz	1.25 MHz
1	1	$f_{PRS}/2^5$	62.5 kHz	156.25 kHz	250 kHz	312.5 kHz

MDL04	MDL03	MDL02	MDL01	MDL00	k	5 位计数器输出时钟的选择
0	0	×	×	×	×	禁止设置
0	1	0	0	0	8	$f_{XCLK0}/8$
0	1	0	0	1	9	$f_{XCLK0}/9$
0	1	0	1	0	10	$f_{XCLK0}/10$
•	•	•	•	•	•	•
•	•	•	•	•	•	•
•	•	•	•	•	•	•
•	•	•	•	•	•	•
•	•	•	•	•	•	•
1	1	0	1	0	26	$f_{XCLK0}/26$
1	1	0	1	1	27	$f_{XCLK0}/27$
1	1	1	0	0	28	$f_{XCLK0}/28$
1	1	1	0	1	29	$f_{XCLK0}/29$
1	1	1	1	0	30	$f_{XCLK0}/30$
1	1	1	1	1	31	$f_{XCLK0}/31$

注 **1.** 如果外设硬件时钟（f_{PRS}）是高速系统时钟（f_{XH}）（XSEL ＝ 1），根据供电电压的不同，f_{PRS} 的工作频率也不同。

- V_{DD} = 2.7 至 5.5 V： $f_{PRS} \leq 10$ MHz
- V_{DD} = 1.8 至 2.7 V： $f_{PRS} \leq 5$ MHz

2. 当选择 TM50 输出作为基准时钟时，应注意以下几点。

- 在 TM50 与 CR50 匹配（TMC506 = 0）时计数时钟被清除并启动的模式
  先启动 8 位定时器/事件计数器 50，然后使能定时器 F/F 反转操作（TMC501 = 1）。
- PWM 模式 （TMC506 = 1）
  先启动 8 位定时器/事件计数器 50，然后设置计数时钟，使占空比 = 50%。

不论何种模式，都无需使能（TOE50 = 1）TO50 输出。

注意事项 **1. 在重写 MDL04 至 MDL00 位时，请确保 ASIM0 寄存器的第 6 位（TXE0）和第 5 位（RXE0） = 0。**
2. 波特率的值是 5 位计数器输出时钟除 2。

备注 **1.** f_{XCLK0}：通过 TPS01 和 TPS00 位选择的基准时钟的频率
2. f_{PRS}：外设硬件时钟频率
3. k：由 MDL04 至 MDL00 位设置的值（k = 8，9，10，...，31）
4. ×：无须理会
5. TMC506：8 位定时器模式控制寄存器 50（TMC50）的第 6 位
TMC501：TMC50 的第 1 位

（4） 功能寄存器 1（PF1）

该寄存器用于设置 P13/SO10/TxD0 引脚的功能。
可以通过 1 位或 8 位存储器操作指令来设置 PF1。
复位信号的产生会将 PF1 设置为 00H。

图 14-5. 端口功能寄存器 1（PF1）的格式

地址：FF20H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
PF1	0	0	0	0	PF13	0	0	0

PF13	端口（P13）、CSI10、UART0和UART6 输出说明
0	用作 P13 或 SO10
1	用作 TxD0或TxD6

（5）　端口模式寄存器 1 （PM1）

该寄存器用于按位设置端口 1 的输入/输出模式。

当 P13/SO10/TxD0/<TxD6>引脚用于串行接口数据输出时，将 PM13 清除为 0。此时 P13 输出锁存器的值可能是 0 或 1。

当 P12/SI10/RxD0/<RxD6>引脚用于串行接口数据输入时，将 PM12 设置为 1。此时 P12 输出锁存器的值可能是 0 或 1。

可以通过 1 位或 8 位存储器操作指令来设置 PM1。

复位信号的产生会将该寄存器设置为 FFH。

图 14-6. 端口模式寄存器 1（PM1）的格式

地址： FF21H　复位后： FFH　R/W

符号	7	6	5	4	3	2	1	0
PM1	1	1	1	PM14	PM13	PM12	PM11	1

PM1n	P1n 引脚 I/O 模式选择（n = 1 至 4）
0	输出模式（输出缓冲器打开）
1	输入模式（输出缓冲器关闭）

14.4 串行接口UART0 的操作

串行接口 UART0 有以下两种模式。

- 操作停止模式
- 异步串行接口（UART）模式

14.4.1 操作停止模式

该模式下，不执行串行通信，可以降低功耗。此外，在该模式下，引脚可作为普通端口引脚使用。要设置操作停止模式，将 ASIM0 的第 7 位、第 6 位和第 5 位（POWER0、TXE0 和 RXE0）清除为 0。

（1） 使用的寄存器

由异步串行接口操作模式寄存器 0（ASIM0）设置操作停止模式。

可以通过 1 位或 8 位存储器操作指令来设置 ASIM0。

复位信号的产生会将该寄存器设置为 01H。

地址： FF70H 复位后： 01H R/W

符号	<7>	<6>	<5>	4	3	2	1	0
ASIM0	POWER0	TXE0	RXE0	PS01	PS00	CL0	SL0	1

POWER0	内部操作时钟的使能/禁止操作
0 [注 1]	禁止内部操作时钟的操作（时钟固定为低电平）并异步复位内部电路[注 2]

TXE0	使能/禁止发送
0	禁止发送（同步复位发送电路）

RXE0	使能/禁止接收
0	禁止接收（同步复位接收电路）

注 **1.** 当 POWER0 = 0 时，从 RxD0 引脚的输入固定为高电平。

2. 异步串行接口接收错误状态寄存器 0（ASIS0）、发送移位寄存器 0（TXS0）和接收缓冲寄存器 0（RXB0）被复位。

注意事项 将 TXE0 和 RXE0 清除为 0 后，再将 POWER0 清除为 0，可以设置操作停止模式。
要启动通信，设置 POWER0 为 1，然后设置 TXE0 或 RXE0 为 1。

备注 要将 RxD0/SI10/<RxD6>/P12 和 TxD0/SO10/<TxD6>/P13 作为通用端口引脚使用，参见**第四章　端口功能**。

14.4.2 异步串行接口（UART）模式

该模式下，在起始位之后，发送/接收 1 字节数据，可以执行全双工操作。

内置专用 UART 波特率发生器，于是可以在较大的波特率范围内执行通信操作。

（1） 使用的寄存器

- 异步串行接口操作模式寄存器 0（ASIM0）
- 异步串行接口接收错误状态寄存器 0（ASIS0）
- 波特率发生器控制寄存器 0（BRGC0）
- 端口模式寄存器 1（PM1）
- 端口寄存器 1（P1）

在 UART 模式下设置操作的基本过程如下。

<1>　设置 BRGC0 寄存器 （参见 图 **14-4**）.
<2>　设置 ASIM0 的第 1 位至第 4 位 （SL0、CL0、PS00 和 PS01）（参见 图 **14-2**）。
<3>　将 ASIM0 的第 7 位（POWER0）设置为 1。
<4>　将 ASIM0 的第 6 位 （TXE0）设置为 1。 →使能发送。
　　　将 ASIM0 的第 5 位 （RXE0）设置为 1。 →使能接收。
<5>　向 TXS0 寄存器写入数据。 → 启动数据发送。

注意事项　在设置端口模式寄存器和端口寄存器时，要考虑与通信另一方的关系。

寄存器设置与引脚之间的关系如下所示。

表 14-2　寄存器设置与引脚之间的关系

POWER0	TXE0	RXE0	PM13	P13	PM12	P12	UART0 操作	引脚功能	
								TxD0/SO10 /<TxD6>/P13	RxD0/SI10 /<RxD6>/P12
0	0	0	×[注]	×[注]	×[注]	×[注]	停止	SO10/<TxD6>/P13	SI10/<RxD6>/P12
1	0	1	×[注]	×[注]	1	×	接收	SO10/P13	RxD0
	1	0	0	×	×[注]	×[注]	发送	TxD0	SI10/P12
	1	1	0	×	1	×	发送/接收	TxD0	RxD0

注　　可以设置为端口，设置为串行接口 CSI10，或设置为串行接口 UART6（只有当 UART0 被停止时）。

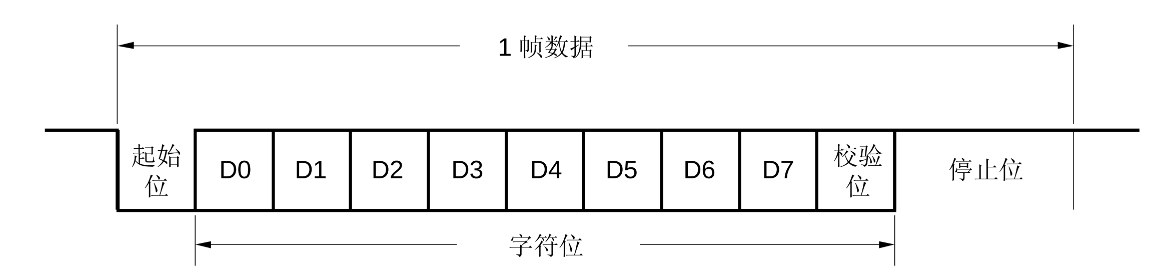
备注　×：　无须理会
　　POWER0：　异步串行接口操作模式寄存器 0（ASIM0）的第 7 位
　　TXE0：　ASIM0 的第 6 位
　　RXE0：　ASIM0 的第 5 位
　　PM1×：　端口模式寄存器
　　P1×：　端口输出锁存器

（**2**）　通信操作

（**a**）正常发送/接收数据的格式和波形示例

图 14-7 和 14-8 是正常发送/接收数据的格式和波形示例。

图 14-7. 正常 UART 发送/接收数据的格式



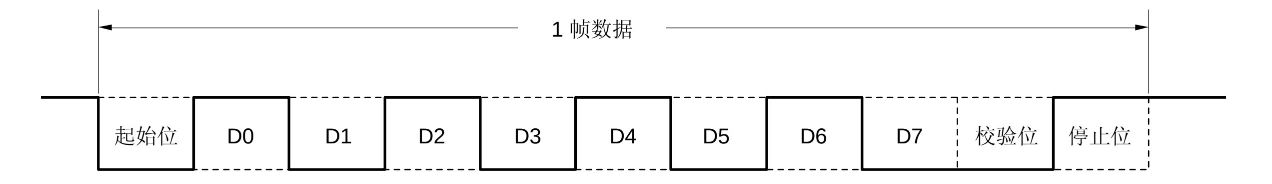
一个数据帧由下列各位组成。

- 起始位 ... 1 位
- 字符位 ... 7 或 8 位 （LSB-先行）
- 校验位 ... 偶校验、奇校验、零校验或无校验
- 停止位 ... 1 或 2 位

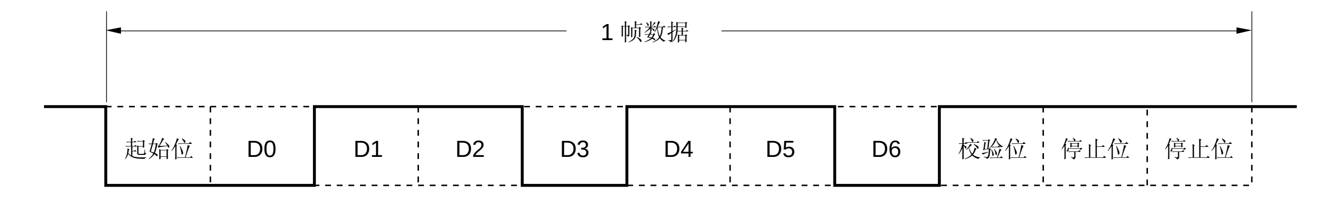
由异步串行接口操作模式寄存器 0（ASIM0）指定一个数据帧中字符位的长度、校验方式和停止位的长度。

图 14-8. 正常 UART 发送/接收数据波形的示例

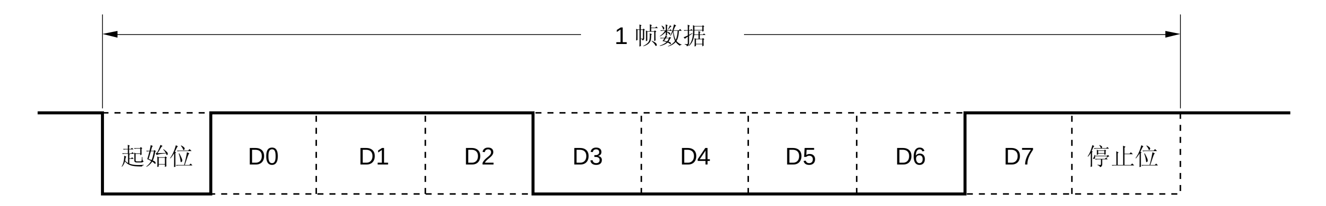
1. 数据长度： **8** 位， 校验位： 偶校验， 停止位： **1** 位， 通信数据： **55H**



2. 数据长度： **7** 位， 校验位： 奇校验， 停止位： **2** 位，通信数据： **36H**



3. 数据长度： **8** 位， 校验位： 无， 停止位： **1** 位，通信数据： **87H**



（b） 校验方式与操作

校验位用于检测数据通信中的位错误。通常，在发送端和接收端都使用相同的校验类型。当采用偶校验和奇校验时，可检测到 1 位（奇数）错误。而采用零校验和无校验时，检测不到错误。

（i） 偶校验

- 发送

  控制发送数据，包括校验位，使得位数据中"1"的数量为偶数。
  校验位的取值如下。

  如果发送数据有奇数个 "1"： 1
  如果发送数据有偶数个 "1"： 0

- 接收

  计算接收数据（包括校验位）中"1"的数量。如果结果为奇数，则产生校验错误。

（ii） 奇校验

- 发送

  与偶校验方式不同，控制发送数据，包括校验位，使得位数据中"1"的数量为奇数。

  如果发送数据有奇数个 "1"： 0
  如果发送数据有偶数个 "1"： 1

- 接收

  计算接收数据中"1"的数量，包括校验位。如果结果为偶数，则产生校验错误。

（iii） 零校验

无论发送数据为何值，发送数据时，校验位被清除为 0。
接收数据时，不检测校验位。因此无论校验位是"0"或"1"，都不会产生校验错误。

（iv） 无校验

没有校验位附加到发送数据。
接收数据时，接收的执行认为接收数据中没有校验位。因为没有校验位，也不会产生校验错误。

（c）发送

如果异步串行接口操作模式寄存器 0（ASIM0）的第 7 位（POWER0）被设置为 1，且 ASIM0 的第 6 位（TXE0）被设置为 1 时，使能发送。将待发送的数据写入发送移位寄存器 0（TXS0），可启动发送操作。起始位、校验位和停止位会自动附加到数据中。

启动发送时，先从 TxD0 引脚输出起始位，然后从 LSB 开始依次发送数据其它位。当发送完成时，由 ASIM0 设置的校验位和停止位被附加到数据中，并产生发送完成中断请求（INTST0）。

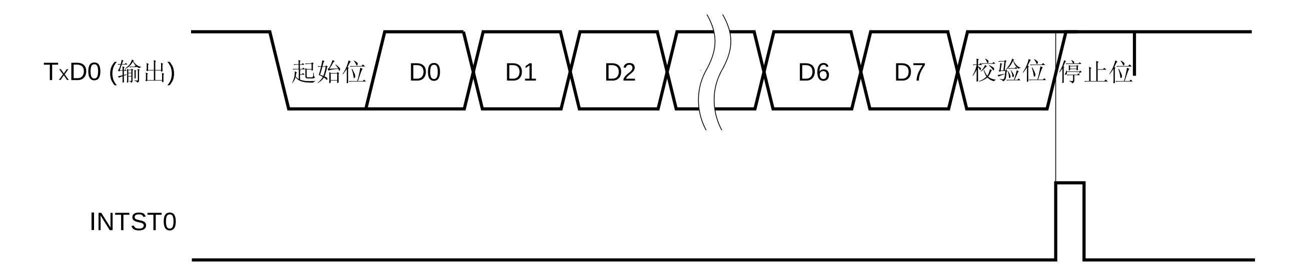
当下一个待发送数据写入 TXS0 时，发送操作被停止。

图 14-9 显示了发送完成中断请求（INTST0）的时序。在最后一个停止位输出后就马上产生该中断。

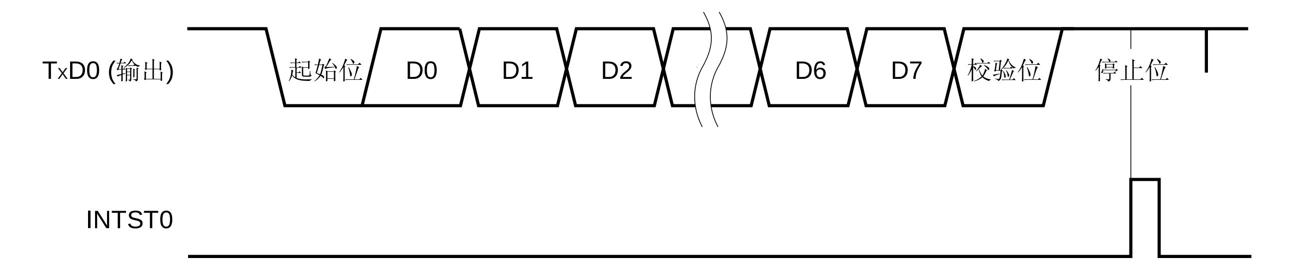
注意事项　待发送数据被写入 TXS0 后，在发送完成中断信号（INTST0）产生之前，不要将下一个发送数据写入 TXS0。

图 **14-9.** 发送完成中断请求的时序

1. 停止位长度：**1**



2. 停止位长度：**2**



（**d**） 接收

当异步串行接口操作模式寄存器 0（ASIM0）的第 7 位（POWER0）被设置为 1，然后将 ASIM0 的第 5 位（RXE0）设置为 1 时，使能接收，并对 RxD0 引脚输入进行采样。

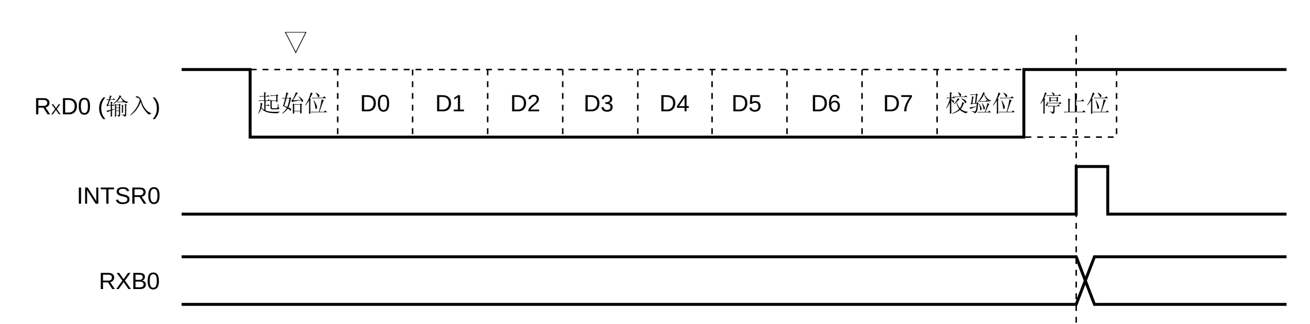
当检测到 RxD0 引脚输入的下降沿时，波特率发生器的 5 位计数器开始计数。当计数值达到波特率发生器控制寄存器 0（BRGC0）的设置值时，再次采样 RxD0 引脚输入 （如图 14-9 中的 ▽）。如果此时 RxD0 为低电平，则输入信号被认为是起始位。

当检测到起始位时，启动接收，按照设置的波特率将串行数据顺序存入接收移位寄存器 0（RXS0）。当接收到停止位时，会产生接收完成中断信号（INTSR0），并且 RXS0 的数据被写入接收缓冲寄存器 0（RXB0）。但是，如果发生溢出错误，则接收数据不写入 RXB0。

即使在接收进行时发生校验错（PE0），接收仍然继续，直至接收到停止位，而在接收完成后，会产生一个接收错误中断（INTSR0）。

当接收操作完成时且存在接收错误的情况下，则产生 INTSR0。

图 14-10. 接收完成中断请求时序



注意事项 **1.** **如果发生接收错误，先读取异步串行接口接收错误状态寄存器 0（ASIS0），然后再读取接收缓冲寄存器 0（RXB0）的内容，可以清除错误标志。否则，当接收到下一个数据时会产生溢出错误，且接收错误状态保持不变。**

2. **总是按照 “停止位的数量 = 1”来执行接收，第 2 个停止位被忽略。**

（e） 接收错误

在接收期间可能发生三种类型的错误：校验错误、帧错误或溢出错误。作为数据接收的结果，如果异步串行接口接收错误状态寄存器 0（ASIS0）的错误标志位被设置，则将会产生一个接收错误中断请求信号（INTSR0）。

在接收错误中断服务（INTSR0）中读取 ASIS0 的内容，可以识别接收期间发生了哪种错误（参见**图 14-3**）。

读取 ASIS0 后，ASIS0 的内容被清除为 0。

表 14-3. 接收错误的原因

接收错误	原因
校验错误	发送数据的校验位与接收数据的校验位不匹配
帧错误	未检测到停止位
溢出错误	在从接收缓冲寄存器 0（RXB0）读取数据之前，已经完成下一个数据的接收

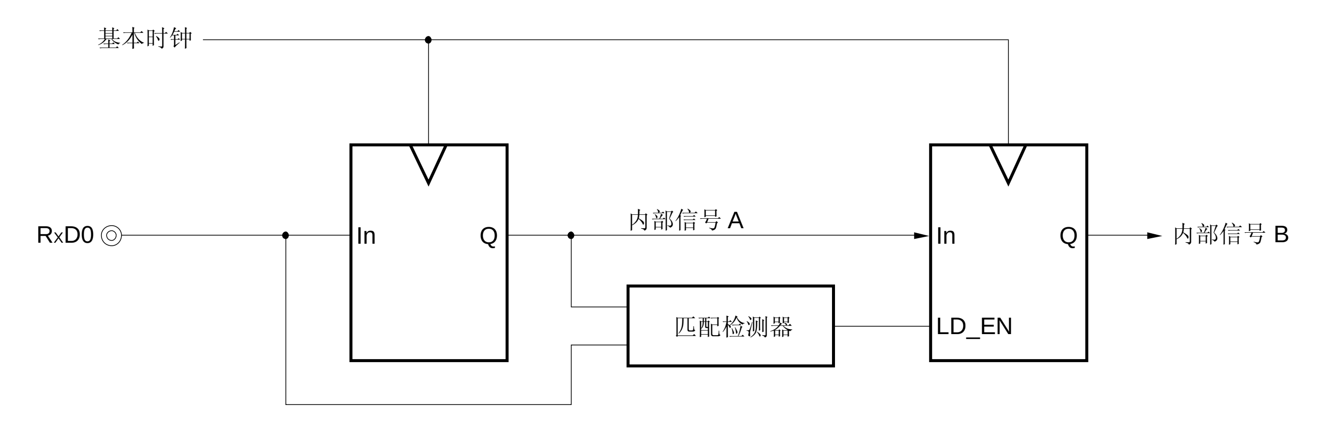
（f） 接收数据的噪音过滤器

通过预分频器设置的基准时钟输出，采样 RxD0 信号。

如果两次采样值相同，则匹配检测器的输出会改变，且被采样的数据作为输入数据。

噪音过滤器的电路结构如图 14-11 所示，外部信号状态之后，延迟两个时钟执行接收操作的内部处理。

图 14-11. 噪音过滤器电路



14.4.3 专用波特率发生器

专用波特率发生器由一个源时钟选择器和一个 5 位可编程计数器组成，用于为 UART0 发送/接收产生一个串行时钟。

为发送和接收提供不同的 5 位计数器。

（1） 波特率发生器的配置

- 基准时钟

  该时钟由波特率发生器控制寄存器 0（BRGC0）的第 7 位和第 6 位（TPS01 和 TPS00）选择。当异步串行接口操作模式寄存器 0（ASIM0）的第 7 位（POWER0）为 1 时，该时钟提供给各个模块。该时钟称为基准时钟，它的频率被称为 f_{XCLK0}。当 POWER0 = 0 时，基准时钟被固定为低电平。

- 发送计数器

  当异步串行接口操作模式寄存器 0（ASIM0）的第 7 位（POWER0）或第 6 位（TXE0）为 0 时，该计数器停止计数，并被清除为 0。

  当 POWER0 = 1 且 TXE0 = 1 时，计数器开始计数。

  当第 1 个待发送数据被写入发送移位寄存器 0（TXS0）时，计数器被清除为 0。

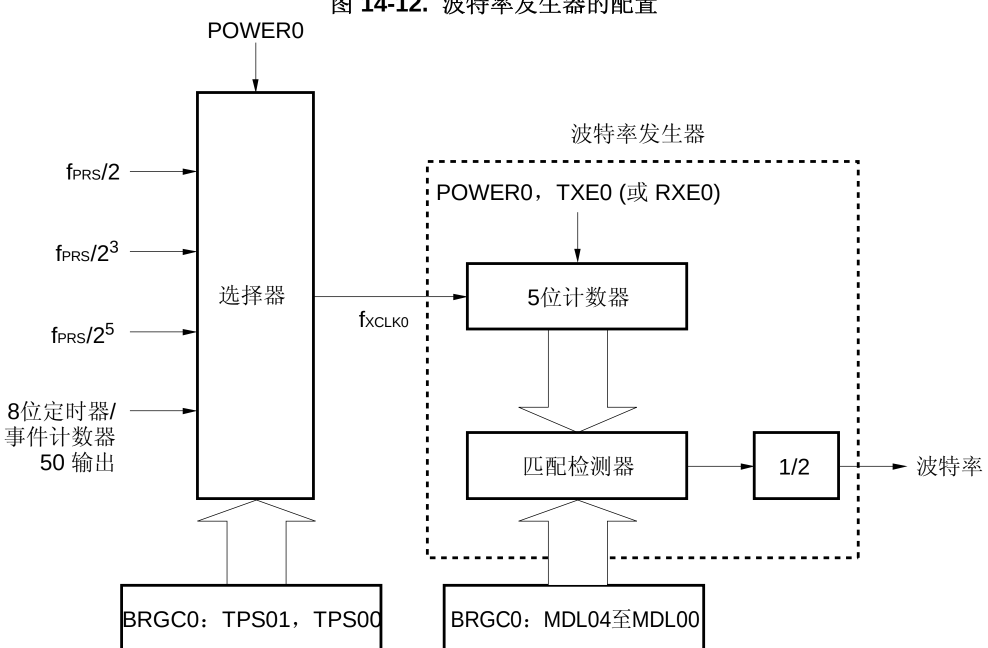
- 接收计数器

  当异步串行接口操作模式寄存器 0（ASIM0）的第 7 位（POWER0）或第 5 位（RXE0）为 0 时，该计数器停止计数，并被清除为 0。

  当检测到起始位时，计数器开始计数。

  接收到一帧数据后，计数器停止计数，直到检测到下一个起始位。

图 14-12. 波特率发生器的配置



备注 POWER0： 异步串行接口操作模式寄存器 0 （ASIM0）的第 7 位
TXE0： ASIM0 的第 6 位
RXE0： ASIM0 的第 5 位
BRGC0： 波特率发生器控制寄存器 0

（2） 串行时钟的产生

使用波特率发生器控制寄存器 0（BRGC0）可产生指定的串行时钟。

使用 BRGC0 的第 7 位和第 6 位（TPS01 和 TPS00），可以选择输入到 5 位计数器的时钟。

BRGC0 的第 4 位至第 0 位（MDL04 至 MDL00）可以用来选择 5 位计数器的分频值。

14.4.4 波特率的计算

（1） 波特率计算表达式

通过下列表达式计算波特率。

- 波特率 = $\frac{f_{XCLK0}}{2 \times k}$ [bps]

f_{XCLK0}： 通过 BRGC0 的 TPS01 和 TPS00 位选择的基准时钟的频率

k： 通过 BRGC0 的 MDL04 至 MDL00 位设置的值 （k = 8，9，10，...，31）

表 14-4. TPS01 和 TPS00 的设置值

TPS01	TPS00	基准时钟 （f_{XCLK0}） 的选择注 1				
			f_{PRS} = 2 MHz	f_{PRS} = 5 MHz	f_{PRS} = 8 MHz	f_{PRS} = 10 MHz
0	0	TM50 输出注 2				
0	1	$f_{PRS}/2$	1 MHz	2.5 MHz	4 MHz	5 MHz
1	0	$f_{PRS}/2^3$	250 kHz	625 kHz	1 MHz	1.25 MHz
1	1	$f_{PRS}/2^5$	62.5 kHz	156.25 kHz	250 kHz	312.5 kHz

注 **1.** 如果外设硬件时钟（f_{PRS}）使用高速系统时钟（f_{XH}）（XSEL = 1），根据供电电压的不同，f_{PRS} 的工作频率也不同。
- V_{DD} = 2.7 至 5.5 V： $f_{PRS} \leq 10$ MHz
- V_{DD} = 1.8 至 2.7 V： $f_{PRS} \leq 5$ MHz

2. 当选择 TM50 输出作为基准时钟时，应注意以下几点。
- 在 TM50 与 CR50 匹配（TMC506 = 0）时计数时钟被清除并启动的模式
  先启动 8 位定时器/事件计数器 50，然后使能定时器 F/F 反转操作（TMC501 = 1）。
- PWM 模式 （TMC506 = 1）
  先启动 8 位定时器/事件计数器 50，然后设置计数时钟，使占空比 = 50%。

不论何种模式，都无需使能（TOE50 = 1）TO50 输出。

（2） 波特率误差

通过下列公式计算波特率误差。

- 误差（%） = $\left(\frac{\text{实际波特率 (有误差的波特率)}}{\text{预期波特率 (正确的波特率)}} - 1\right) \times 100$ [%]

注意事项 1. 在发送期间，必须保持波特率误差在接收目的方允许误差范围内。

2. 在接收期间，波特率的误差必须满足“（4）接收期间允许的波特率范围”的范围。

举例：　基准时钟的频率= 2.5 MHz = 2,500,000 Hz

BRGC0 的 MDL04 至 MDL00 位的设置值 = 10000B （k = 16）

目标波特率 = 76，800 bps

目标波特率 = 2.5 M/（2 × 16）

= 2,500,000/（2 × 16） = 78,125 [bps]

误差 = （78,125/76,800 − 1） × 100

= 1.725 [%]

（3） 波特率设置的示例

表 14-5. 波特率发生器的数据设置

	波特率 [bps]	f_{PRS} = 2.0 MHz				f_{PRS} = 5.0 MHz				f_{PRS} = 10.0 MHz			
		TPS01, TPS00	k	计算值	ERR [%]	TPS01, TPS00	k	计算值	ERR [%]	TPS01, TPS00	k	计算值	ERR [%]
	1200	3H	26	1202	0.16	–	–	–	–	–	–	–	–
	2400	3H	13	2404	0.16	–	–	–	–	–	–	–	–
	4800	2H	26	4808	0.16	3H	16	4883	1.73	–	–	–	–
	9600	2H	13	9615	0.16	3H	8	9766	1.73	3H	16	9766	1.73
	10400	2H	12	10417	0.16	2H	30	10417	0.16	3H	15	10417	0.16
	19200	1H	26	19231	0.16	2H	16	19531	1.73	3H	8	19531	1.73
	24000	1H	21	23810	–0.79	2H	13	24038	0.16	2H	26	24038	0.16
	31250	1H	16	31250	0	2H	10	31250	0	2H	20	31250	0
<R>	33600	1H	15	33333	–0.79	2H	9	34722	3.34	2H	19	32895	–2.1
	38400	1H	13	38462	0.16	2H	8	39063	1.73	2H	16	39063	1.73
	56000	1H	9	55556	–0.79	1H	22	56818	1.46	2H	11	56818	1.46
	62500	1H	8	62500	0	1H	20	62500	0	2H	10	62500	0
	76800	–	–	–	–	1H	16	78125	1.73	2H	8	78125	1.73
	115200	–	–	–	–	1H	11	113636	–1.36	1H	22	113636	–1.36
	153600	–	–	–	–	1H	8	156250	1.73	1H	16	156250	1.73
<R>	312500	–	–	–	–	–	–	–	–	1H	8	312500	0

备注　TPS01，TPS00：波特率发生器控制寄存器 0(BRGC00)的第 7 位和第 6 位（基准时钟（f_{XCLK0}）的设置）

k：　根据 BRGC0 的 MDL04 至 MDL00 位设置的值 （k = 8，9，10，...，31）

f_{PRS}：　外设硬件时钟频率

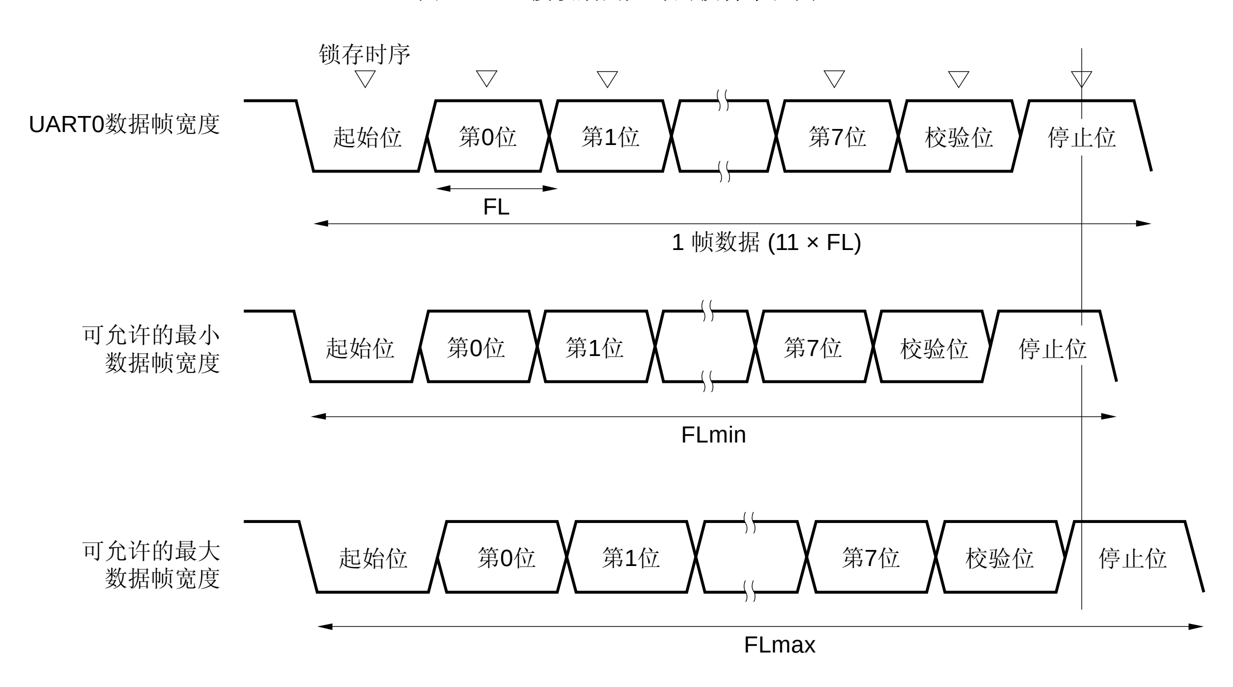
ERR：　波特率误差

（4） 接收期间允许的波特率范围

接收期间，发送目的方的波特率允许误差范围如下所示。

注意事项　接收期间，必须确保波特率误差在允许的误差范围内，可以使用以下表达式计算。

图 14-13. 接收期间允许的波特率范围



如图 14-13 所示，当检测到起始位后，接收数据的锁存时序由波特率发生器控制寄存器 0（BRGC0）设置的计数器决定。如果最后的数据（停止位）满足该锁存时序，则该数据可以正确接收。

假定接收到的数据是 11 位，理论值可以计算如下。

FL = （Brate）$^{-1}$

Brate：　UART0 的波特率

k：　BRGC0 的设置值

FL：　1 位数据长度

锁存时序的余量：　2 个时钟

可允许的最小数据帧长度： $FLmin = 11 \times FL - \frac{k-2}{2k} \times FL = \frac{21k+2}{2k} FL$

因此，发送目的方可以接收的最大波特率如下所示。

$$BRmax = (FLmin/11)^{-1} = \frac{22k}{21k+2} Brate$$

同样，允许的最大数据帧长度可以计算如下。

$$\frac{10}{11} \times FLmax = 11 \times FL - \frac{k+2}{2 \times k} \times FL = \frac{21k-2}{2 \times k} FL$$

$$FLmax = \frac{21k-2}{20k} FL \times 11$$

因此，发送目的方可以接收的最小波特率如下所示。

$$BRmin = (FLmax/11)^{-1} = \frac{20k}{21k-2} Brate$$

通过上述最小和最大波特率表达式，可以计算 UART0 与发送目的方之间允许的波特率误差，如下所示。

表 14-6. 允许的最大/最小波特率误差

分频比 （k）	允许的最大波特率误差	允许的最小波特率误差
8	+3.53%	–3.61%
16	+4.14%	–4.19%
24	+4.34%	–4.38%
31	+4.44%	–4.47%

备注
1. 接收的允许误差取决于一帧的位数量、输入时钟频率和分频比（k）。输入时钟频率越高，且分频比（k）越高，允许的误差就越大。
2. k： BRGC0 的设置值。

15.1 串行接口UART6 的功能

串行接口 UART6 有以下两种模式。

（1）　操作停止模式

不执行串口通信时，使用该模式，可以降低功耗。

详情参见 **15.4.1 操作停止模式。**

（2）　异步串行接口（UART） 模式

该模式支持 LIN（局域内联网）总线。该模式的功能概括如下。

详情参见 **15.4.2 异步串行接口（UART）模式** 和 **15.4.3 专用波特率发生器。**

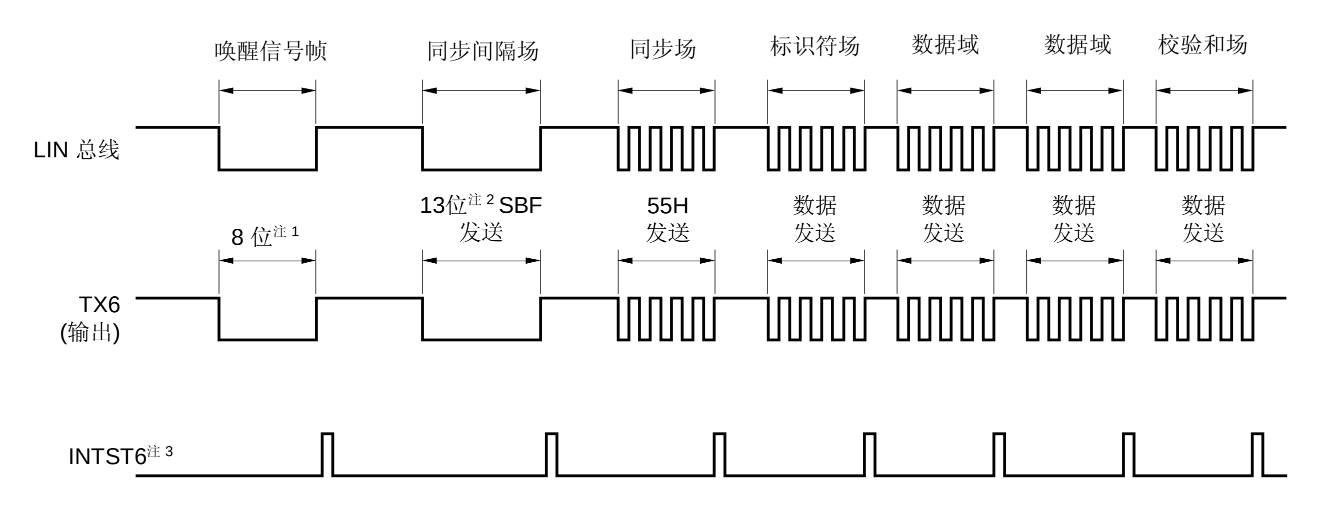
- 最大传输率：　625 kbps
- 双引脚配置　TxD6：　发送数据输出引脚
RxD6：　接收数据输入引脚
- TxD6/RxD6 引脚可以通过寄存器选择使用 P112/P113（默认）或 P13/P12。
- 通信数据的长度可以选择 7 位或 8 位。
- 专用内部 8 位波特率发生器，允许设置任意波特率。
- 发送和接收可以独立执行（全双工操作）。
- 可以选择 MSB-先行或 LSB-先行通信。
- 反向发送操作。
- 同步间隔场发送长度为 13 至 20 位。
- 可以识别 11 位以上的同步间隔场接收（提供 SBF 接收标志位）。

注意事项
1. **TxD6** 输出反向功能仅在发送端反向而不在接收端反向。要使用该功能，接收端应准备好接收已反向的数据。
2. 如果提供给串行接口 **UART6** 的时钟未停止（例如 **HALT** 模式下），则继续正常操作。如果提供的时钟已经停止（例如 **STOP** 模式下），则每个寄存器都停止操作，并且保持时钟停止前的瞬时值。**TxD0** 引脚也保持时钟停止前的瞬时值，并输出该电平。但是，在时钟供应恢复之后，该操作无法保证。因此，需要复位该电路，使得 **POWER6 = 0**、**RXE6 = 0** 且 **TXE6 = 0**。
3. 设置 **POWER6 = 1**，然后设置 **TXE6 = 1**（发送）或 **RXE6 = 1**（接收），开始通信。
4. 通过 **CKSR6** 设置的基准时钟（f_{XCLK6}）使 **TXE6** 与 **RXE6** 同步。要再次使能发送或接收，在 **TXE6** 或 **RXE6** 被清除为 **0** 后，至少经过两个基准时钟，再将 **TXE6** 或 **RXE6** 设置为 **1**。如果在两个基准时钟时以内对 **TXE6** 或 **RXE6** 置位，则发送电路或接收电路可能无法初始化。
5. 设置 **TXE6 = 1** 之后，经过至少一个基准时钟，才能将发送数据设置到 **TXB6**。
6. 如果连续发送数据，从停止位到下一个起始位的通信时序，要延长到该模块的两个操作时钟。但这不会影响到通信结果，因为接收端会在检测到起始位时初始化时序。如果该接口用于 **LIN** 通信操作，则不要使用连续发送功能。

备注 LIN 表示局域内联网，是一个低速（1 至 20 kbps）串行通信协议，可以降低汽车网络的费用。
LIN 通信是单主通信，一个主设备上最多可以连接 15 个从设备。
LIN 的从设备用于控制开关、制动器和传感器，这些设备通过 LIN 网络连接到 LIN 主设备。
通常，LIN 主设备连接到一个网络，比如 CAN（Controller Area Network）。
此外，LIN 总线采用单线方式，通过符合 ISO9141 的收发器与各节点相连。
在 LIN 协议中，主设备发送带有波特率信息的一帧数据，从设备接收到该数据并调整波特率误差。因此当从设备端的波特率误差在 ±15%范围内时，可以进行通信。

图 15-1 和 15-2 概括了 LIN 的发送和接收操作。

图 15-1 LIN 发送操作



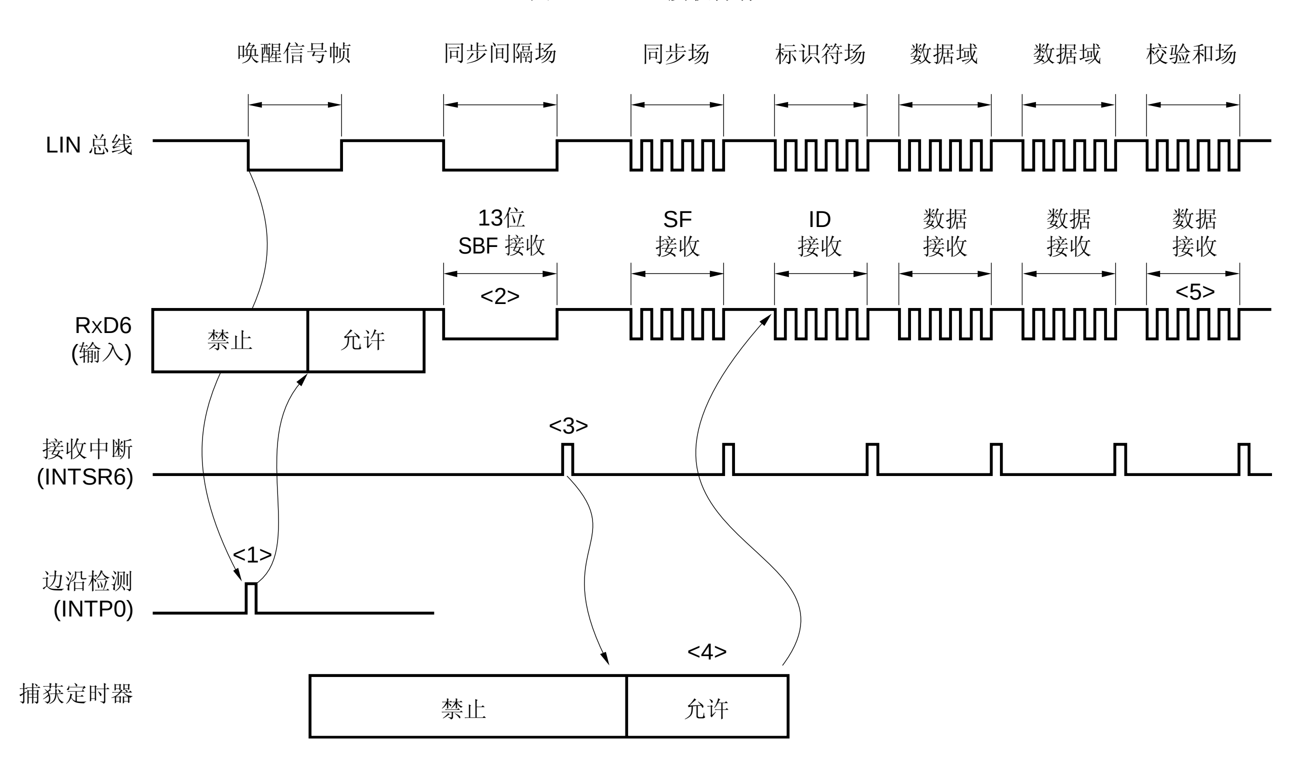
注 **1.** 在 8 位模式下，可通过发送 80H 替代唤醒信号帧。

2. 使用硬件输出同步间隔场。输出宽度就是位长度，可以通过异步串行接口控制寄存器 6（ASICL6）的第 4 位至第 2 位（SBL62 至 SBL60）设置（参见 **15.4.2 （2） （h） SBF** 发送）。

3. 每次发送完成时输出 INTST6。在发送 SBF 时也会输出 INTST6。

备注 各个场之间的间隔由软件控制。

图 15-2 LIN 接收操作



接收处理过程如下。

<1> 在引脚的边沿处检测唤醒信号，使能 UART6，并设置 SBF 接收模式。

<2> 检测到停止位之前，持续接收。在检测到一个 SBF 具有 11 位或更多的低电平数据时，认为 SBF 已经完成正确接收，并输出一个中断信号。如果检测到的 SBF 具有少于 11 位的低电平数据时，认为发生 SBF 接收错误。不输出中断信号，且恢复 SBF 接收模式。

<3> 如果 SBF 已经完成正确接收，将输出一个中断信号。通过这个 SBF 接收完成中断服务来启动 16 位定时器/事件计数器 00，并测量同步场的位间隔（脉冲宽度）（参见 **6.4.8 脉冲宽度测量操作**）。对 OVE6、PE6 和 FE6 错误的检测被暂停，不执行 UART 通信的错误检测处理，也不执行移位寄存器和 RXB6 之间的数据传送。移位寄存器中保存复位值 FFH。

<4> 由同步场的位间隔计算波特率误差，在 SBF 接收后禁止 UART6，然后对波特率发生器控制寄存器 6（BRGC6）重新设置。

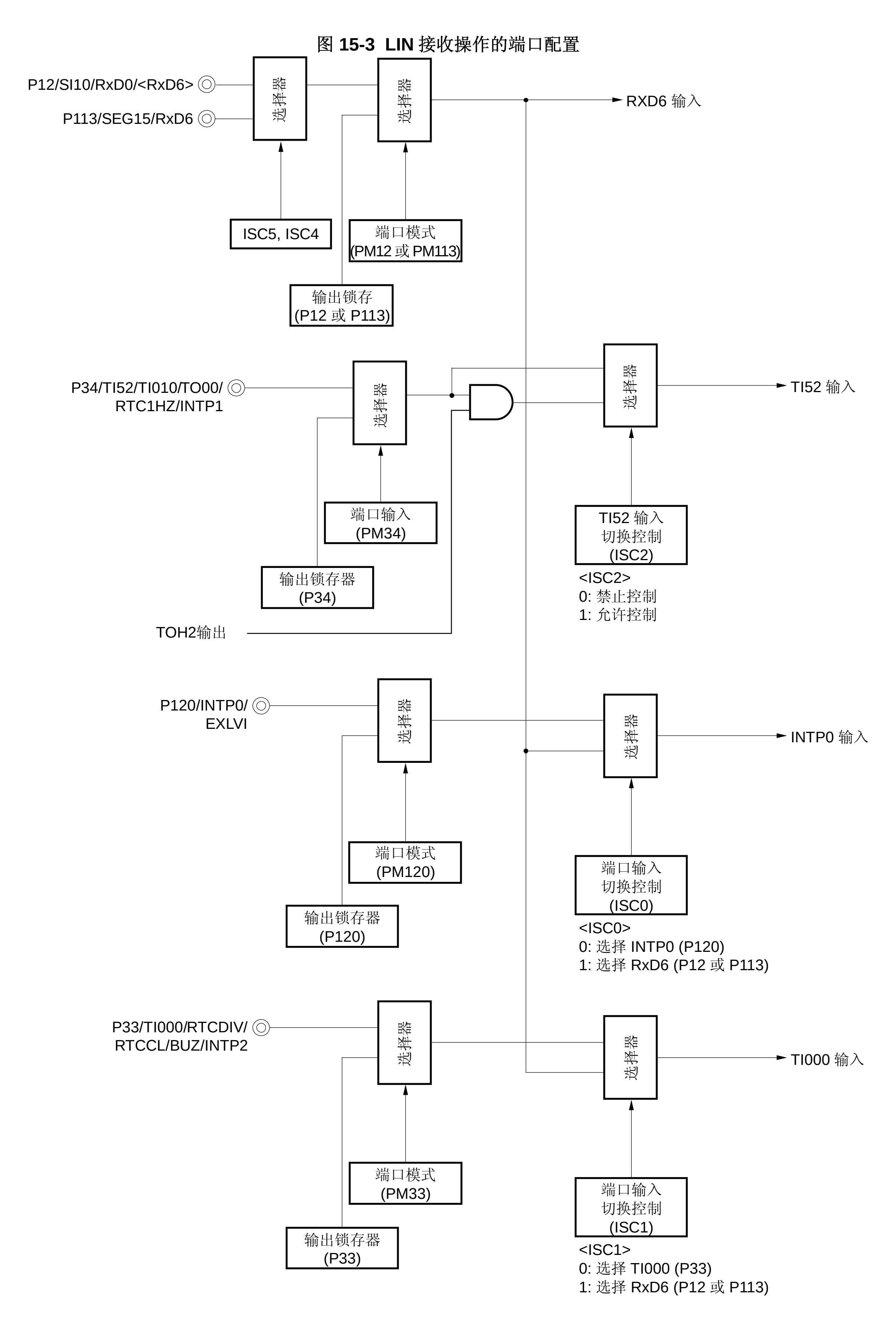
<5> 通过软件区分校验和场。在接收到校验和场后，通过软件对 UART6 执行初始化处理，并再次设置 SBF 接收模式。

图 15-3 显示了 LIN 接收操作的端口配置。

通过检测外部中断（INTP0）边沿，接收从 LIN 主设备发出的唤醒信号。使用 16 位定时器/事件计数器 00 的外部事件捕获操作，可以测量 LIN 主设备发送的同步场长度，也可以计算波特率误差。

无需外部连接 RxD6 和 INTP0/TI000，由端口输入切换控制（ISC0/ISC1）就可以将接收端口输入（RxD6）输入到外部中断（INTP0）和 16 位定时器/事件计数器 00。

图 15-3 LIN 接收操作的端口配置



备注 ISC0，ISC1，ISC2，ISC4，ISC5：输入切换控制寄存器（ISC）的第 0 位，第 1 位，第 2 位，第 4 位和第 5 位（参见 **图 15-11**）。

以下是 LIN 通信中使用的外设功能。

<使用的外设功能>

- 外部中断（INTP0）；唤醒信号检测
  用途： 检测唤醒信号边沿，并检测通信的启动。
- 16 位定时器/事件计数器 00（TI000）；波特率误差检测
  用途： 通过检测同步场（SF）长度并根据位的数量平分，可以检测波特率误差（在捕获模式下测量 TI000 输入边沿的间隔）。
- 串行接口 UART6

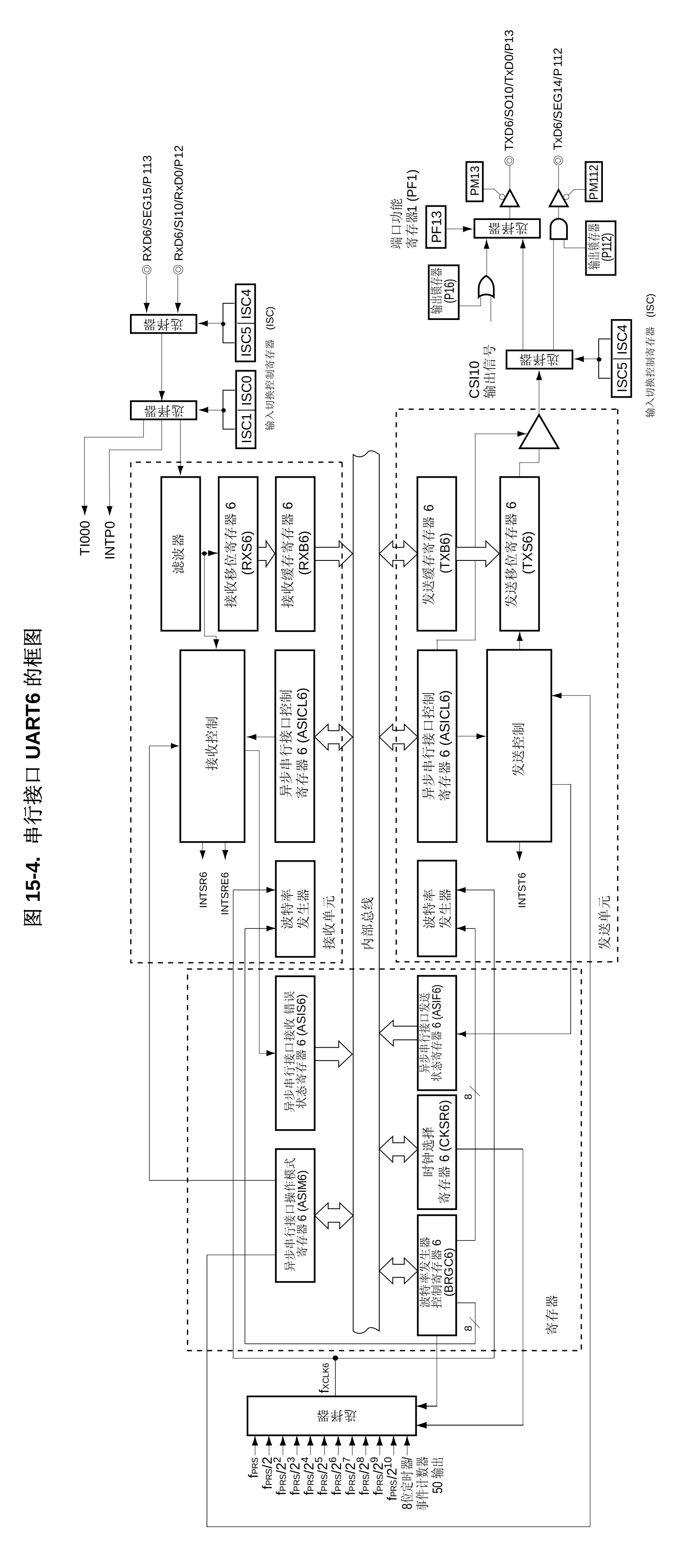
15.2 串行接口UART6 的配置

串行接口 UART6 包括以下硬件。

表 15-1 串行接口 UART6 的配置

项目	配置
寄存器	接收缓冲寄存器 6 （RXB6） 接收移位寄存器 6 （RXS6） 发送缓冲寄存器 6 （TXB6） 发送移位寄存器 6 （TXS6）
控制寄存器	异步串行接口操作模式寄存器 6 （ASIM6） 异步串行接口接收错误状态寄存器 6 （ASIS6） 异步串行接口发送状态寄存器 6 （ASIF6） 时钟选择寄存器 6 （CKSR6） 波特率发生器控制寄存器 6 （BRGC6） 异步串行接口控制寄存器 6 （ASICL6） 输入切换控制寄存器 （ISC） 端口功能寄存器 1 （PF1） 端口模式寄存器 1 （PM1） 端口寄存器 1 （P1） 端口模式寄存器 11 （PM11） 端口寄存器 11 （P11）

图 15-4. 串行接口 UART6 的框图



（1） 接收缓冲寄存器 6（RXB6）

该 8 位寄存器用于存储由接收移位寄存器（RXS6）转换的并行数据。

每接收到 1 字节数据，新的接收数据就从 RXS6 传送到该寄存器中。如果数据长度被设定为 7 位，则按如下方式传送数据。

- 在 LSB-先行接收时，接收到的数据传送到 RXB6 的第 0 位至第 6 位，而 RXB6 的 MSB 位恒为 0。
- 在 MSB-先行接收时，接收到的数据传送到 RXB6 的第 1 位至第 7 位，而 RXB6 的 LSB 位恒为 0。

如果发生溢出错误（OVE6），则不把接收数据传送到 RXB6。

可以通过 8 位存储器操作指令来读取 RXB6。不能将数据写入该寄存器。

复位信号的产生会将该寄存器设置为 FFH。

（2） 接收移位寄存器 6（RXS6）

该寄存器将输入到 RxD6 引脚的串行数据转换为并行数据。

程序不能直接操作 RXS6。

（3） 发送缓冲寄存器 6（TXB6）

该寄存器用于设置发送数据。当数据写入 TXB6 时，启动发送。

可以通过 8 位存储器操作指令对该寄存器进行读或写操作。

复位信号的产生会将该寄存器设置为 FFH。

注意事项 1. 当异步串行接口发送状态寄存器 6（ASIF6）的第 1 位（TXBF6）为 1 时，不要向 TXB6 写入数据。

2. 在通信过程中（当异步串行接口操作模式寄存器 6（ASIM6）的第 7 位（POWER6）和第 6 位（TXE6）均为 1，或 ASIM6 的第 7 位（POWER6）和第 5 位（RXE6）均为 1），不要通过软件刷新（写入相同值）TXB6。

3. 在设置 TXE6 = 1 之后，至少经过一个基准时钟（f_{XCLK6}），将发送数据设置到 TXB6。

（4） 发送移位寄存器 6（TXS6）

该寄存器将 TXB6 中待传送的数据作为串行数据，从 TxD6 引脚传输。在第 1 次发送时，当数据被写入 TXB6 后，立即从 TXB6 传输，或者在持续发送时，传输 1 帧数据后，紧邻在产生 INTST6 中断之前数据被传送。在基准时钟的下降沿将来自 TXB6 的数据从 TxD6 引脚发送。

程序不能直接操作 TXS6。

15.3 控制串行接口UART6 的寄存器

串行接口 UART6 由以下 12 个寄存器控制。

- 异步串行接口操作模式寄存器 6（ASIM6）
- 异步串行接口接收错误状态寄存器 6（ASIS6）
- 异步串行接口发送状态寄存器 6（ASIF6）
- 时钟选择寄存器 6（CKSR6）
- 波特率发生器控制寄存器 6（BRGC6）
- 异步串行接口控制寄存器 6（ASICL6）
- 输入切换控制寄存器（ISC）
- 端口功能寄存器 1（PF1）
- 端口模式寄存器 1（PM1）
- 端口寄存器 1（P1）
- 端口模式寄存器 11（PM11）
- 端口寄存器 11（P11）

（1） 异步串行接口操作模式寄存器 6（ASIM6）

该 8 位寄存器用于控制串行接口 UART6 的串行通信操作。

可以通过 1 位或 8 位存储器操作指令来设置该寄存器。

复位信号的产生会将该寄存器设置为 01H。

备注 在通信操作期间（当异步串行接口操作模式寄存器 6（ASIM6）的第 7 位（POWER6）和第 6 位（TXE6）= 1，或 ASIM6 的第 7 位（POWER6）和第 5 位（RXE6）= 1），可以通过软件刷新（写入相同的值）ASIM6。

图 15-5 异步串行接口操作模式寄存器 6（ASIM6）的格式（1/2）

地址：FF50H 复位后： 01H R/W

符号	<7>	<6>	<5>	4	3	2	1	0
ASIM6	POWER6	TXE6	RXE6	PS61	PS60	CL6	SL6	ISRM6

POWER6	使能/禁止内部操作时钟的操作
0[注1]	禁止内部操作时钟的操作（时钟固定为低电平）并异步复位内部电路[注2]
1	使能内部操作时钟的操作

TXE6	使能/禁止发送
0	禁止发送（同步复位发送电路）
1	使能发送

RXE6	使能/禁止接收
0	禁止接收（同步复位接收电路）
1	使能接收

注 **1.** 发送期间，当 POWER6 = 0 时，TxD6 引脚的输出变为高电平，而从 RxD6 引脚的输入被固定为高电平。

2. 异步串行接口接收错误状态寄存器 6（ASIS6）、异步串行接口发送状态寄存器 6（ASIF6）、异步串行接口控制寄存器 6（ASICL6）的第 7 位（SBRF6）和第 6 位（SBRT6），以及接收缓冲寄存器 6（RXB6）被复位。

图 15-5 异步串行接口操作模式寄存器 6（ASIM6）的格式（2/2）

PS61	PS60	发送操作	接收操作
0	0	不输出校验位	无校验接收
0	1	输出零校验	按零校验接收[注]
1	0	输出奇校验	按奇校验判断
1	1	输出偶校验	按偶校验判断

CL6	指定发送/接收数据的字符长度
0	数据的字符长度 = 7 位
1	数据的字符长度 = 8 位

SL6	指定发送数据停止位的数量
0	停止位的数量 = 1
1	停止位的数量 = 2

ISRM6	出错的情况下使能/禁止接收完成中断的产生
0	出错时产生“INTSRE6”（此时，不产生 INTSR6）
1	出错时产生“INTSR6”（此时，不产生 INTSRE6）

注　如果选择“按零校验接收”，则不判断校验。因此异步串行接口接收错误状态寄存器 0（ASIS6）的第 2 位（PE6）不被置位，这样就不会产生错误中断。

注意事项
1. **要启动发送，将 POWER6 设置为 1，然后将 TXE6 设置为 1。要停止发送，先将 TXE6 清除为 0，然后将 POWER6 清除为 0。**
2. **要启动接收，将 POWER6 设置为 1，然后将 RXE6 设置为 1。要停止接收，先将 RXE6 清除为 0，然后将 POWER6 清除为 0。**
3. **当 RxD6 引脚输入为高电平时，先将 POWER6 设置为 1，然后将 RXE6 设置为 1。当输入为低电平时设置 POWER6 为 1，且设置 RXE6 为 1 时，可以启动接收。**
4. **通过 CKSR6 设置的基准时钟（f_{XCLK6}）使 TXE6 与 RXE6 同步。要再次使能发送或接收，在 TXE6 或 RXE6 被清除为 0 后，至少经过两个基准时钟，再将 TXE6 或 RXE6 设置为 1。如果在两个基准时钟时以内对 TXE6 或 RXE6 置位，则发送电路或接收电路可能无法初始化。**
5. **设置 TXE6 = 1 之后，经过至少一个基准时钟（f_{XCLK6}），才能将发送数据设置到 TXS6。**
6. **在重写 PS61、PS60 和 CL6 位之前，先将 TXE6 和 RXE6 位清除为 0。**
7. **用于 LIN 通信操作时，PS61 与 PS60 位固定为 0。**
8. **重写 SL6 位时，必须确保 TXE6 = 0。总是按照“停止位的数量=1”执行接收操作，因此，不会受到 SL6 设置值的影响。**
9. **重写 ISRM6 位时，请确保 RXE6 = 0。**

（2） 异步串行接口接收错误状态寄存器 6（ASIS6）

该寄存器用于表示串行接口 UART6 接收操作完成时的错误状态，包括 3 个错误标志位（PE6，FE6，OVE6）。

只能通过 8 位存储器操作指令来读取该寄存器。

复位信号的产生，或将 ASIM6 的第 5 位（RXE6）或第 7 位（POWER6）清除为 0，都会将该寄存器清除为 00H。读取该寄存器时，读到的值是 00H。如果产生一个接收错误，则先读取 ASIS6 然后再读取接收缓冲寄存器 6（RXB6），可以清除错误标志。

图 15-6 异步串行接口接收错误状态寄存器 6（ASIS6）的格式

地址： FF53H 复位后： 00H R

符号	7	6	5	4	3	2	1	0
ASIS6	0	0	0	0	0	PE6	FE6	OVE6

PE6	表示校验错误的状态标志位
0	如果 POWER6 = 0 且 RXE6 = 0，或 ASIS6 寄存器被读取
1	如果发送数据的校验位和接收完成时的校验位不匹配

FE6	表示帧错误的状态标志位
0	如果 POWER6 = 0 且 RXE6 = 0，或 ASIS6 寄存器被读取
1	如果在接收完成时未检测到停止位

OVE6	表示溢出错误的状态标志位
0	如果 POWER6 = 0 且 RXE6 = 0，或 ASIS6 寄存器被读取
1	如果接收数据被设置在 RXB6 寄存器中，并且在读取该数据之前，下一个接收操作已完成

注意事项 **1. 根据异步串行接口操作模式寄存器 6（ASIM6）的 PS61 和 PS60 位的设置值，PE6 位的操作也有所不同。**

2. 无论停止位的数量如何设置，只能检测接收数据的第 1 位作为停止位。

3. 如果发生溢出错误，则下一个接收数据不被写入接收缓冲寄存器 6（RXB6），而是被丢弃。

4. 如果从 ASIS6 读取数据，会产生一个等待周期。当 CPU 运行于副系统时钟操作且外设硬件时钟处于停止状态时，不要从 ASIS6 读取数据。详情参见第三十三章 等待注意事项。

（3） 异步串行接口发送状态寄存器 6（ASIF6）

该寄存器表示串行接口 UART6 的发送状态，包括两个状态标志位（TXBF6 和 TXSF6）。

当数据从 TXB6 寄存器传送到 TXS6 寄存器之后，将下一个待发送数据写入 TXB6 寄存器，可以连续发送而不被打断，即使在中断期间也不受影响。

只能通过 8 位存储器操作指令来读取该寄存器。

复位信号的产生，或将 ASIM6 的第 7 位（POWER6）或第 6 位（TXE6）清除为 0，都会将该寄存器清除为 00H。

图 15-7 异步串行接口发送状态寄存器 6（ASIF6）的格式

地址： FF55H 复位后： 00H R

符号	7	6	5	4	3	2	1	0
ASIF6	0	0	0	0	0	0	TXBF6	TXSF6

TXBF6	发送缓冲数据标志位
0	如果 POWER6 = 0 或 TXE6 = 0，或如果数据被传送到发送移位寄存器 6（TXS6）
1	当数据被写入发送缓冲寄存器 6（TXB6）中（如果数据已存在于 TXB6）

TXSF6	发送移位寄存器数据标志位
0	如果 POWER6 = 0 或 TXE6 = 0，或如果在发送完成后下一个数据没有从发送缓冲寄存器 6（TXB6）传输
1	如果数据从发送缓冲寄存器 6（TXB6）传输（如果数据发送进行中）

注意事项 1. 要连续发送数据，应将第一个发送的数据（第一个字节）写入 TXB6 寄存器。请确保检查 TXBF6 标志位为“0”。如果正确，将下一个发送数据（第二个字节）写入 TXB6 寄存器。如果在 TXBF6 标志为“1”时将数据写入 TXB6，则发送数据无法保证。

2. 要在连续发送完成时对发送单元初始化，则请确保在产生发送完成中断后检查 TXSF6 标志位为“0”，然后执行初始化操作。如果在 TXSF6 标志为“1”时执行初始化操作，则发送数据无法保证。

（4） 时钟选择寄存器 6（CKSR6）

该寄存器用于选择串行接口 UART0 的基准时钟。

可以通过 8 位存储器操作指令来设置 CKSR6。

复位信号的产生会将该寄存器清除为 00H。

备注 在通信操作期间（当异步串行接口操作模式寄存器 6（ASIM6）的第 7 位（POWER6）和第 6 位（TXE6）= 1，或 ASIM6 的第 7 位（POWER6）和第 5 位（RXE6）= 1），可以通过软件刷新（写入相同的值）CKSR6。

图 15-8 时钟选择寄存器 6（CKSR6）的格式

地址： FF56H 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
CKSR6	0	0	0	0	TPS63	TPS62	TPS61	TPS60

TPS63	TPS62	TPS61	TPS60	基准时钟（f_{XCLK6}） 的选择[注 1]				
					f_{PRS} = 2 MHz	f_{PRS} = 5 MHz	f_{PRS} = 8 MHz	f_{PRS} = 10 MHz
0	0	0	0	f_{PRS}[注 2]	2 MHz	5 MHz	8 MHz	10 MHz
0	0	0	1	$f_{PRS}/2$	1 MHz	2.5 MHz	4 MHz	5 MHz
0	0	1	0	$f_{PRS}/2^2$	500 kHz	1.25 MHz	2 MHz	2.5 MHz
0	0	1	1	$f_{PRS}/2^3$	250 kHz	625 kHz	1 MHz	1.25 MHz
0	1	0	0	$f_{PRS}/2^4$	125 kHz	312.5 kHz	500 kHz	625 kHz
0	1	0	1	$f_{PRS}/2^5$	62.5 kHz	156.25 kHz	250 kHz	312.5 kHz
0	1	1	0	$f_{PRS}/2^6$	31.25 kHz	78.13 kHz	125 kHz	156.25 kHz
0	1	1	1	$f_{PRS}/2^7$	15.625 kHz	39.06 kHz	62.5 kHz	78.13 kHz
1	0	0	0	$f_{PRS}/2^8$	7.813 kHz	19.53 kHz	31.25 kHz	39.06 kHz
1	0	0	1	$f_{PRS}/2^9$	3.906 kHz	9.77 kHz	15.625 kHz	19.53 kHz
1	0	1	0	$f_{PRS}/2^{10}$	1.953 kHz	4.88 kHz	7.513 kHz	9.77 kHz
1	0	1	1	TM50 输出[注 3]				
其它情况				禁止设置				

注 1. 如果外设硬件时钟（f_{PRS}）是高速系统时钟（f_{XH}）（XSEL = 1），根据供电电压的不同，f_{PRS} 的工作频率也不同。

- V_{DD} = 2.7 至 5.5 V： $f_{PRS} \leq 10$ MHz
- V_{DD} = 1.8 至 2.7 V： $f_{PRS} \leq 5$ MHz

2. 如果外部硬件时钟 （f_{PRS}） 是内部高速振荡时钟（f_{RH}）（XSEL = 0），当 1.8 V ≤ V_{DD} < 2.7 V 时，禁止设定 TPS63 = TPS62 = TPS61 = TPS60 = 0 （基准时钟： f_{PRS}）。

3. 当选择 TM50 输出作为基准时钟时，应注意以下几点。

- 在 TM50 与 CR50 匹配（TMC506 = 0）时计数时钟被清除并启动的模式
  先启动 8 位定时器/事件计数器 50，然后使能定时器 F/F 反转操作（TMC501 = 1）。
- PWM 模式 （TMC506 = 1）
  先启动 8 位定时器/事件计数器 50，然后设置计数时钟，使占空比 = 50%。

不论何种模式，都无需使能（TOE50 = 1）TO50 输出。

注意事项 当重写 TPS63 至 TPS60 位时，请确保 POWER6 = 0。

备注 **1.** f_{PRS}： 外设硬件时钟频率
2. TMC506： 8 位定时器模式控制寄存器 50（TMC50）的第 6 位
TMC501： TMC50 的第 1 位

（5） 波特率发生器控制寄存器 6（BRGC6）

该寄存器用于设置串行接口 UART6 的 8 位计数器的分频值。
可以通过 8 位存储器操作指令来设置 BRGC6。
复位信号的产生会将该寄存器设置为 FFH。

备注 在通信操作期间（当异步串行接口操作模式寄存器 6（ASIM6）的第 7 位（POWER6）和第 6 位（TXE6）= 1，或 ASIM6 的第 7 位（POWER6）和第 5 位（RXE6）= 1），可以通过软件刷新（写入相同的值）BRGC6。

图 15-9. 波特率发生器控制寄存器 6（BRGC6）的格式

地址： FF57H 复位后： FFH R/W

符号	7	6	5	4	3	2	1	0
BRGC6	MDL67	MDL66	MDL65	MDL64	MDL63	MDL62	MDL61	MDL60

MDL67	MDL66	MDL65	MDL64	MDL63	MDL62	MDL61	MDL60	k	8 位计数器的输出时钟选择
0	0	0	0	0	0	×	×	×	禁止设置
0	0	0	0	0	1	0	0	4	f_{XCLK6}/4
0	0	0	0	0	1	0	1	5	f_{XCLK6}/5
0	0	0	0	0	1	1	0	6	f_{XCLK6}/6
•	•	•	•	•	•	•	•	•	•
1	1	1	1	1	1	0	0	252	f_{XCLK6}/252
1	1	1	1	1	1	0	1	253	f_{XCLK6}/253
1	1	1	1	1	1	1	0	254	f_{XCLK6}/254
1	1	1	1	1	1	1	1	255	f_{XCLK6}/255

注意事项 1. 当重写 MDL67 至 MDL60 位时，请确保 ASIM0 的第 6 位（TXE6）和第 5 位（RXE6） = 0。
2. 波特率的值是 8 位计数器输出时钟除 2。

备注 **1.** f_{XCLK6}： 由 CKSR6 寄存器的 TPS63 至 TPS60 位选择的基准时钟的频率
2. k： 由 MDL67 至 MDL60 位设置的值 （k = 4，5，6，...，255）
3. ×： 无须理会

（6） 异步串行接口控制寄存器 6（ASICL6）

该寄存器用于控制串行接口 UART6 的串行通信操作。

可以通过 1 位或 8 位存储器操作指令来设置 ASICL6。

复位信号的产生会将该寄存器设置为 16H。

注意事项 **在通信操作期间（当异步串行接口操作模式寄存器 6（ASIM6）的第 7 位（POWER6）和第 6 位（TXE6）= 1，或 ASIM6 的第 7 位（POWER6）和第 5 位（RXE6）= 1），可以通过软件刷新（写入相同的值）ASICL6。但是，在 SBF 接收（SBRT6 = 1）期间或 SBF 发送（直到由于 SBTT6 被置位（1）而产生 INTST6 为止）期间，不要将 SBRT6 与 SBTT6 都设置为 1，因为这样可能会重新触发 SBF 接收和 SBF 发送。**

图 15-10 异步串行接口控制寄存器 6（ASICL6）的格式（1/2）

地址： FF58H 复位后： 16H R/W[注]

符号	<7>	<6>	5	4	3	2	1	0
ASICL6	SBRF6	SBRT6	SBTT6	SBL62	SBL61	SBL60	DIR6	TXDLV6

SBRF6	SBF 接收状态标志
0	如果 POWER6 = 0 且 RXE6 = 0，或者如果 SBF 接收已经正确完成
1	SBF 接收进行中

SBRT6	SBF 接收触发
0	–
1	SBF 接收触发

SBTT6	SBF 发送触发
0	–
1	SBF 发送触发

注 第 7 位是只读的。

图 15-10 异步串行接口控制寄存器 6（ASICL6）的格式（2/2）

SBL62	SBL61	SBL60	SBF 发送输出宽度控制
1	0	1	SBF 输出宽度为 13 位
1	1	0	SBF 输出宽度为 14 位
1	1	1	SBF 输出宽度为 15 位
0	0	0	SBF 输出宽度为 16 位
0	0	1	SBF 输出宽度为 17 位
0	1	0	SBF 输出宽度为 18 位
0	1	1	SBF 输出宽度为 19 位
1	0	0	SBF 输出宽度为 20 位

DIR6	首位说明
0	MSB
1	LSB

TXDLV6	使能/禁止反向 TxD6 输出
0	TxD6 的正常输出
1	TxD6 的反向输出

注意事项
1. 在 **SBF** 接收错误的情况下，返回到 **SBF** 接收模式，**SBRF6** 标志位的状态被保持（**1**）。
2. 在设置 **SBRT6** 位之前，请确保 **ASIM6** 的第 **7** 位（**POWER6**）和第 **5** 位（**RXE6**）**= 1**。在设置 **SBRT6** 位为 **1** 之后，在 **SBF** 接收完成之前（产生中断请求信号之前），不要将其清除为 **0**。
3. **SBRT6** 位的读取值恒为 **0**。在 **SBF** 接收正确完成后，**SBRT6** 被自动清除为 **0**。
4. 在设置 **SBTT6** 位为 **1** 之前，请确保 **ASIM6** 的第 **7** 位（**POWER6**）和第 **6** 位（**TXE6**）**= 1**。在设置 **SBTT6** 位为 **1** 之后，在 **SBF** 接收完成之前（产生中断请求信号之前），不要将其清除为 **0**。
5. **SBTT6** 位的读取值恒为 **0**。在 **SBF** 发送结束时，**SBRT6** 被自动清除为 **0**。
6. 在接收期间，不要设置 **SBRT6** 位为 **1**，而在发送期间，不要设置 **SBTT6** 位为 **1**。
7. 重写 **DIR6** 和 **TXDLV6** 位之前，将 **TXE6** 和 **RXE6** 清除为 **0**。

（7） 输入切换控制寄存器（ISC）

通过设定 ISC5 为 1，UART6 I/O 引脚从 P113/SEG15SEG15/RxD6 和 P112/SEG14SEG14/TxD6 切换为 P12/SI10/RxD0/<RxD6> 和 P13/SO10/TxD0/<TxD6>。

通过设定 ISC3 为 1，使能 P113/SEG15/RxD6 引脚作为输入。当 ISC3 被清除为 0 时，不响应外部输入。因此，在复位释放后，可以避免由于不确定的输入状态而产生电流流过此引脚，直到执行输出设置。

在 LIN（局域内联网）接收期间，输入切换控制寄存器（ISC）用于接收从主设备发送的状态信号。

通过设置 ISC0 和 ISC1 为 1，切换 P12/SI10/RxD0/<RxD6>或 P113/SEG15/RxD6 引脚输入的信号作为 INTP0 和 TI000 的输入源。

可以通过 1 位或 8 位存储器操作指令来设置该寄存器。

复位信号的产生会将该寄存器清除为 00H。

图 15-11. 输入切换控制寄存器（ISC）的格式

地址： FF4FH 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
ISC	0	0	ISC5	ISC4	ISC3	ISC2	ISC1	ISC0

ISC5	ISC4	TxD6，RxD6 输入源选择
0	0	TxD6：P112，RxD6： P113
0	1	TxD6：P13，RxD6： P12
其它情况		禁止设置

ISC3	RxD6/P113 输入使能/禁止
0	RxD6/P113 输入禁止
1	RxD6/P113 输入使能

ISC2	TI52 输入源控制
0	不使能 TI52 输入（P34）的控制
1	使能 TI52 输入（P34）的控制[注]

ISC1	TI000 输入源的选择
0	TI000（P33）
1	RxD6（P12 或 P113）

ISC0	INTP0 输入源的选择
0	INTP0（P120）
1	RxD6（P12 或 P113）

注 TI52 输入由 TOH2 输出信号控制。

注意事项 **复位释放后，P113/SEG15/RxD6 引脚被用作 P113 或 RxD6 引脚时，设置 PF11ALL 为 0 且将 ISC3 设置为 1。**

复位释放后，P113/SEG15/RxD6 引脚被用作 SEG15 引脚时，设置 PF11ALL 为 1 且将 ISC3 设置为 0。

（8） 端口功能寄存器 1 （PF1）

该寄存器用于设置 P13/SO10/TxD0/<TxD6>引脚的功能。

可以通过 1 位或 8 位存储器操作指令来设置 PF1。

复位信号的产生会将 PF1 设置为 00H。

图 15-12. 端口功能寄存器 1（PF1）的格式

地址： FF20H 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
PF1	0	0	0	0	PF13	0	0	0

PF13	端口（P13）、CSI10、UART0和UART6 输出说明
0	用作 P13 或SO10
1	用作 TxD0或TxD6

（9） 端口模式寄存器 1（PM1）

该寄存器用于按位设置端口 1 的输入/输出模式。

当 P13/SO10/TXD0/<TXD6> 引脚用于串行接口数据输出时，将 PM13 清除为 0。此时 P13 输出锁存器的值可能是 0 或 1。

当 P12/SI10/RxD0/<RxD6>引脚用于串行接口数据输入时，将 PM12 设置为 1。此时 P12 输出锁存器的值可能是 0 或 1。

可以通过 1 位或 8 位存储器操作指令来设置 PM1。

复位信号的产生会将该寄存器设置为 FFH。

图 15-13. 端口模式寄存器 1（PM1）的格式

地址： FF21H 复位后： FFH R/W

符号	7	6	5	4	3	2	1	0
PM1	1	1	1	PM14	PM13	PM12	PM11	1

PM1n	P1n 引脚 I/O 模式选择（n = 1 至 4）
0	输出模式 （输出缓冲器打开）
1	输入模式 （输出缓冲器关闭）

（10） 端口模式寄存器 11（PM11）

该寄存器用于按位设置端口 1 的输入/输出模式。

当 P112/SEG14/TxD6 引脚用于串行接口数据输出时，将 PM112 清除为 0。此时 P112 输出锁存器的值可能是 0 或 1。

当 P113/SEG15/RxD6 引脚用于串行接口数据输入时，将 PM113 设置为 1。此时 P113 输出锁存器的值可能是 0 或 1。

可以通过 1 位或 8 位存储器操作指令来设置 PM11。

复位信号的产生会将该寄存器设置为 FFH。

图 15-14. 端口模式寄存器 11 （PM11） 的格式

地址： FF2BH　复位后： FFH　R/W

符号	7	6	5	4	3	2	1	0
PM11	1	1	1	1	PM113	PM112	PM111	PM110

PM11n	P11n 引脚 I/O 模式选择（n = 0 至 3）
0	输出模式 （输出缓冲器打开）
1	输入模式 （输出缓冲器关闭）

15.4 串行接口UART6 的操作

串行接口 UART6 有以下两种模式。

- 操作停止模式
- 异步串行接口（UART） 模式

15.4.1 操作停止模式

该模式下，不执行串行通信，可以降低功耗。此外，在该模式下，引脚可作为普通端口引脚使用。要设置操作停止模式，将 ASIM6 的第 7 位、第 6 位和第 5 位（POWER6、TXE6 和 RXE6）清除为 0。

（1） 使用的寄存器

由异步串行接口操作模式寄存器 6（ASIM6）设置操作停止模式。

可以通过 1 位或 8 位存储器操作指令来设置 ASIM6。

复位信号的产生会将该寄存器设置为 01H。

地址： FF50H 复位后： 01H R/W

符号	<7>	<6>	<5>	4	3	2	1	0
ASIM6	POWER6	TXE6	RXE6	PS61	PS60	CL6	SL6	ISRM6

POWER6	内部操作时钟的使能/禁止操作
0 [注 1]	禁止内部操作时钟的操作（时钟固定为低电平）并异步复位内部电路[注 2]

TXE6	使能/禁止发送
0	禁止发送操作（同步复位发送电路）

RXE6	使能/禁止接收
0	禁止接收操作（同步复位接收电路）

注 **1.** 发送期间，当 POWER6 = 0 时，TxD6 引脚的输出变为高电平，而从 RxD6 引脚的输入被固定为高电平。

2. 异步串行接口接收错误状态寄存器 6（ASIS6）、异步串行接口发送状态寄存器 6（ASIF6）、异步串行接口控制寄存器 6（ASICL6）的第 7 位（SBRF6）和第 6 位（SBRT6），以及接收缓冲寄存器 6（RXB6）被复位。

注意事项 **将 TXE6 和 RXE6 清除为 0 后，再将 POWER6 清除为 0，可以停止操作。**
要启动通信，设置 POWER6 为 1，然后设置 TXE6 或 RXE6 为 1。

备注 要将 RxD6/P12 和 TxD6/P13 或 RxD6/P113 和 TxD6/P112 引脚用作通用端口引脚，参见**第四章 端口功能**。

15.4.2 异步串行接口（UART）模式

该模式下，在起始位之后，发送/接收 1 字节数据，可以执行全双工操作。

内置专用 UART 波特率发生器，于是可以在较大的波特率范围内执行通信操作。

（1） 使用的寄存器

- 异步串行接口操作模式寄存器 6 （ASIM6）
- 异步串行接口接收错误状态寄存器 6 （ASIS6）
- 异步串行接口发送状态寄存器 6 （ASIF6）
- 时钟选择寄存器 6 （CKSR6）
- 波特率发生器控制寄存器 6 （BRGC6）
- 异步串行接口控制寄存器 6 （ASICL6）
- 输入切换控制寄存器（ISC）
- 端口模式寄存器 1 （PM1）
- 端口寄存器 1 （P1）
- 端口模式寄存器 11 （PM11）
- 端口寄存器 11 （P11）

在 UART 模式下设置操作的基本过程如下。

<1> 设置 CKSR6 寄存器 （参见图 **15-8**）
<2> 设置 BRGC6 寄存器 （参见图 **15-9**）
<3> 设置 ASIM6 的第 0 位至第 4 位 （ISRM6、SL6、CL6、PS60、PS61）（参见 图 **15-5**）
<4> 设置 ASICL6 的第 0 位和第 1 位 （TXDLV6、DIR6）（参见 图 **15-10**）
<5> 设置 ASIM6 的第 7 位 （POWER6）为 1
<6> 设置 ASIM6 的第 6 位 （TXE6）为 1。 → 使能发送
设置 ASIM6 的第 5 位 （RXE6）为 1。 → 使能接收
<7> 将数据写入发送缓冲寄存器 6 （TXB6） → 开始发送数据

注意事项 在设置端口模式寄存器和端口寄存器时，要考虑与通信另一方的关系。

寄存器设置与引脚之间的关系如下所示。

表 **15-2.** 寄存器设置与引脚之间的关系

（**a**） 通过 **ISC** 寄存器的第 **4** 位，第 **5** 位（**ISC4**，**ISC5**）选择 **P12** 和 **P13** 作为 **UART6** 引脚时

POWER6	TXE6	RXE6	PM13	P13	PM12	P12	UART6 操作	引脚功能	
								TxD6/SO10/TxD0/P13	RxD6/SI10/RxD0/P12
0	0	0	×注	×注	×注	×注	停止	SO10/TxD0/P13	SI10/RxD0/P12
1	0	1	×注	×注	1	×	接收	SO10/P13	RxD6
	1	0	0	×	×注	×注	发送	TxD6	SI10/P12
	1	1	0	×	1	×	发送/接收	TxD6	RxD6

注　可以设置为端口功能，设置为串行接口 CIS10，或设置为串行接口 UART0（只有当 UART6 被停止时）。

注意事项　TxD6/SEG14/P112 和 **RxD6/SEG15/P113** 引脚用作 **SEG14/P112** 和 **SEG15/P113**。

备注　×：　无须理会
POWER6：　异步串行接口操作模式寄存器 6（ASIM6）的第 7 位
TXE6：　ASIM6 的第 6 位
RXE6：　ASIM6 的第 5 位
PM1×：　端口模式寄存器
P1×：　端口输出锁存器

（**b**） 通过 **ISC** 寄存器的第 **4** 位，第 **5** 位（**ISC4**，**ISC5**）选择 **P112** 和 **P113** 作为 **UART6** 引脚时

POWER6	TXE6	RXE6	PM112	P112	PM113	P113	UART6 操作	引脚功能	
								TxD6/SEG14/P112	RxD6/SEG15/P113
0	0	0	×注	×注	×注	×注	停止	SEG14/P112	SEG15/P113
1	0	1	×注	×注	1	×	接收	SEG14/P112	RxD6
	1	0	0	1	×注	×注	发送	TxD6	SEG15/P113
	1	1	0	1	1	×	发送/接收	TxD6	RxD6

注　可以设置为端口功能或 segment 输出。

注意事项　TxD6/SO10/TxD0/P13 和 **RxD6/SI10/RxD0/P12** 引脚用作 **SO10/TxD0/P13** 和 **SI10/RxD0/P12**。

备注　×：　无须理会
POWER6：　异步串行接口操作模式寄存器 6（ASIM6）的第 7 位
TXE6：　ASIM6 的第 6 位
RXE6：　ASIM6 的第 5 位
PM1×：　端口模式寄存器
P1×：　端口输出锁存器

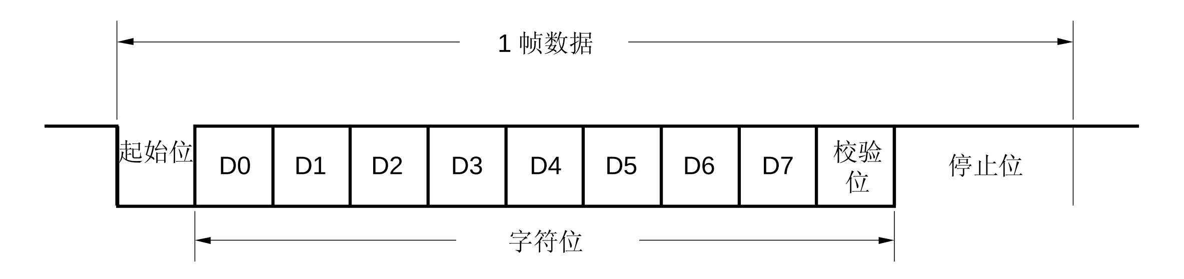
（2） 通信操作

（a） 正常发送/接收数据的格式和波形示例

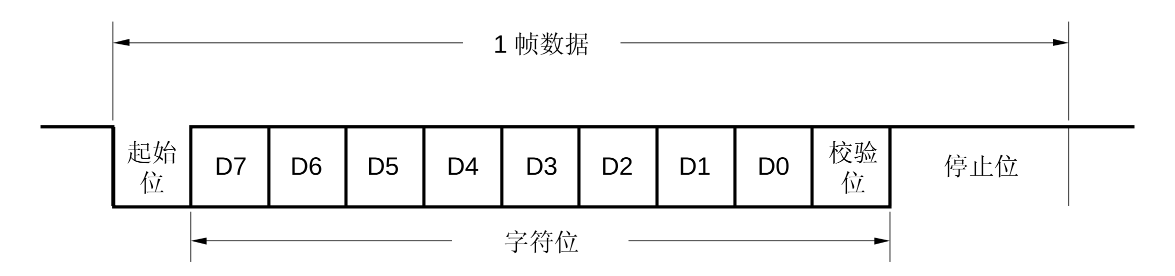
图 15-15 和 15-16 时正常发送/接收数据的格式和波形示例。

图 15-15. 正常 UART 发送/接收数据的格式

1. LSB-先行发送/接收



2. MSB-先行发送/接收



一个帧数据由下列各位组成。

- 起始位 ... 1 位
- 字符位 ... 7 或 8 位
- 校验位 ... 偶校验、奇校验、零校验或无校验
- 停止位 ... 1 或 2 位

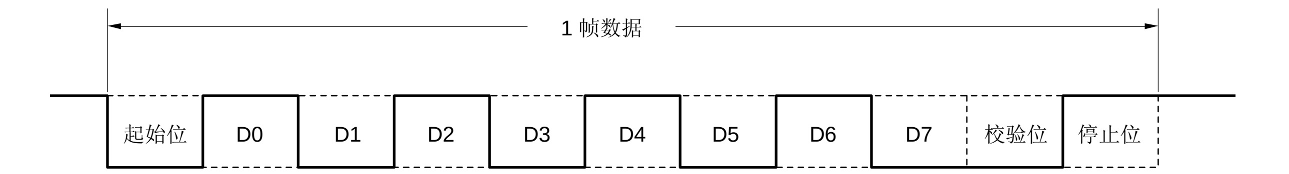
由异步串行接口操作模式寄存器 6（ASIM6）指定一个数据帧中字符位的宽度、校验方式和停止位的宽度。

由异步串行接口控制寄存器 6 （ASICL6）的第 1 位（DIR6）指定数据通信是 LSB-先行或 MSB-先行。

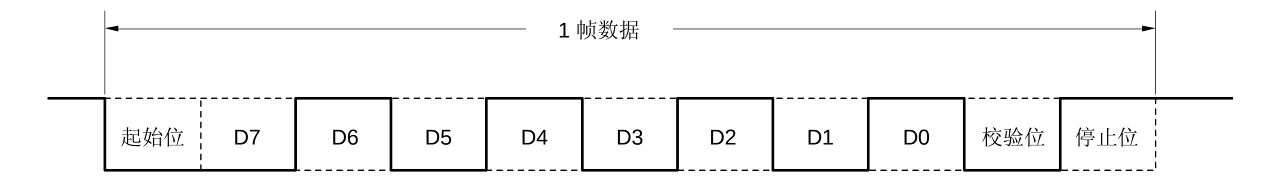
由 ASICL6 的第 0 位（TXDLV6）指定 TxD6 引脚输出的是正常数据或反向数据。

图 15-16. 正常 UART 发送/接收数据波形示例

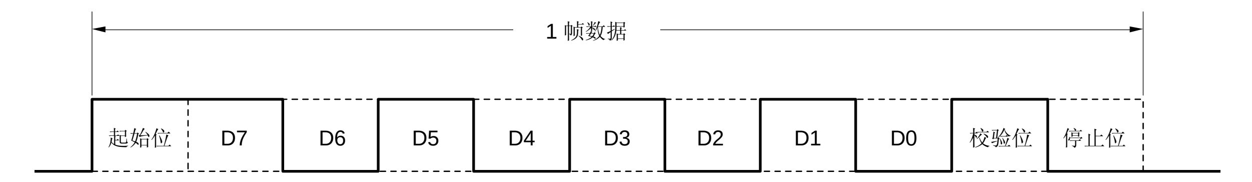
1. 数据长度： **8** 位、**LSB-**先行，校验位： 偶校验， 停止位： **1** 位， 通信数据： **55H**



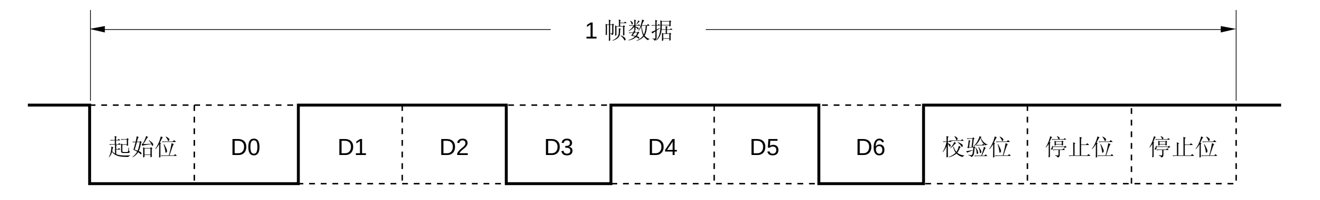
2. 数据长度： **8** 位、**MSB-**先行，校验位： 偶校验， 停止位： **1** 位， 通信数据： **55H**



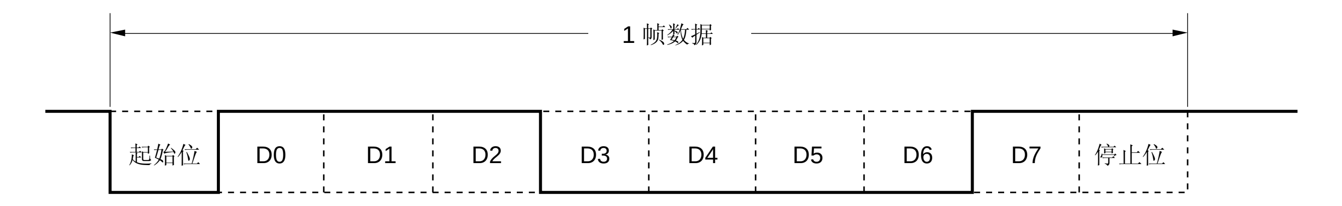
3. 数据长度： **8** 位、**MSB-**先行， 校验位： 偶校验， 停止位： **1** 位， 通信数据： **55H**，**TxD6** 引脚输出反向数据



4. 数据长度： **7** 位、**LSB-**先行， 校验位： 奇校验， 停止位： **2** 位， 通信数据： **36H**



5. 数据长度： **8** 位、**LSB-**先行， 校验位： 无， 停止位： **1** 位， 通信数据： **87H**



（b） 校验方式与操作

校验位用于检测数据通信中的位错误。通常，在发送端和接收端都使用相同的校验类型。当采用偶校验和奇校验时，可检测到 1 位（奇数）错误。而采用零校验和无校验时，检测不到错误。

注意事项 当该设置用于 LIN 通信操作时，PS61 与 PS60 位恒为 0。

（i） 偶校验

- 发送

  控制发送数据，包括校验位，使得数据中“1”的数量为偶数。

  校验位的取值如下。

  如果发送数据有奇数个“1”： 1

  如果发送数据有偶数个“1”： 0

- 接收

  计算接收数据（包括校验位）中 “1”的数量。如果“1”的数量为奇数，则产生校验错。

（ii） 奇校验

- 发送

  与偶校验不同，控制发送数据，包括校验位，使得数据中“1”的数量为奇数。

  如果发送数据有奇数个“1”： 0

  如果发送数据有偶数个“1”： 1

- 接收

  计算接收数据中“1”的数量，包括校验位。如果是偶数，则产生校验错。

（iii） 零校验

无论发送数据为何值，发送数据时，校验位被清除为 0。

接收数据时，不检测校验位。因此无论校验位是“0”或“1”，都不会产生校验错误。

（iv） 无校验

没有校验位附加到发送数据。

接收数据时，接收的执行认为接收数据中没有校验位。因为没有校验位，也不会产生校验错误。

（c） 正常发送

如果异步串行接口操作模式寄存器 6（ASIM6）的第 7 位（POWER6）被设置为 1，且 ASIM6 的第 6 位（TXE6）被设置为 1 时，使能发送。将待发送的数据写入发送移位寄存器 6（TXS6），可启动发送操作。起始位、校验位和停止位会自动附加到数据中。

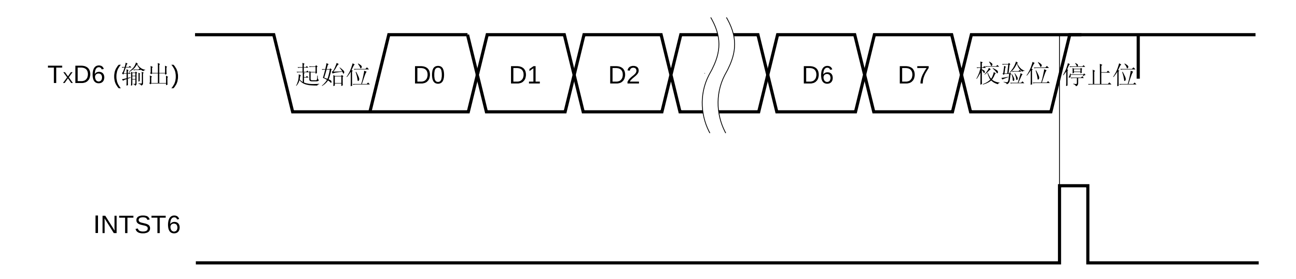
启动发送时，TXB6 中的数据被传送到发送移位寄存器 6（TXS6），然后发送数据依次从 TXS6 输出到 TxD6 引脚。当发送完成时，由 ASIM6 设置的校验位和停止位被附加到数据中，并产生发送完成中断请求（INTST6）。

当下一个待发送数据写入 TXB6 时，发送操作被停止。

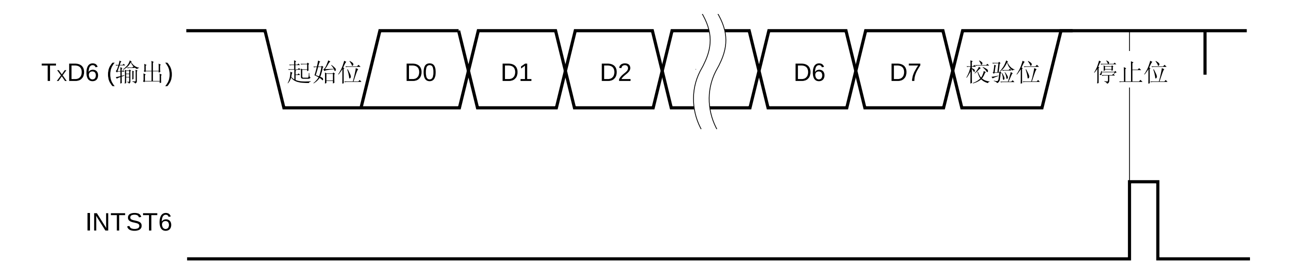
图 15-17 显示了发送完成中断请求（INTST6）的时序。在最后一个停止位输出后就马上产生该中断。

图 15-17. 正常的发送完成中断请求时序

1. 停止位长度： 1



2. 停止位长度： 2



（d） 连续发送

当发送移位寄存器 6 （TXS6）启动移位操作时，马上就可以把下一个发送数据写入发送缓冲寄存器 6 （TXB6）。因此，当发送完一帧数据后，即使在执行 INTST6 中断服务时也能够连续发送数据，从而实现高效率的通信速率。此外，当产生发送完成中断时，通过读取异步串行接口发送状态寄存器 ASIF6 的第 0 位（TXSF6），可以对 TXB6 高效写入两次（2 个字节）而无须等待一帧数据的发送时间。

要连续发送数据，请确保访问 ASIF6 寄存器来检查发送状态，判断是否可以对 TXB6 写入，然后写入数据。

注意事项 1. 在连续发送期间，ASIF6 寄存器的 TXBF6 和 TXSF6 标志从“10” 变为 “11”，再变为 “01”。因此检查状态时，不要使用 TXBF6 和 TXST6 的组合来判断。在连续发送时，只读 TXBF6 标志。

2. 当该设备用于 LIN 通信操作时，不能使用连续发送功能。将发送数据写入发送缓冲寄存器 6（TXB6）之前，请确保异步串行接口发送状态寄存器 6（ASIF6）是 00H。

TXBF6	写入 TXB6 寄存器
0	使能写入
1	禁止写入

注意事项 要连续发送数据，应将第一个发送的数据（第一个字节）写入 TXB6 寄存器。请确保检查 TXBF6 标志位为“0”。如果正确，将下一个发送数据（第二个字节）写入 TXB6 寄存器。如果在 TXBF6 标志为“1”时将数据写入 TXB6，则发送数据无法保证。

可以使用 TXSF6 标志位来检查通信状态。

TXSF6	发送状态
0	发送完成
1	发送进行中

注意事项 1. 要在连续发送完成时对发送单元初始化，请确保在产生发送完成中断后检查 TXSF6 标志位为“0”，然后执行初始化操作。如果在 TXSF6 标志为“1”时执行初始化操作，则发送数据无法保证。

2. 在连续发送期间，当一帧数据发送完成后，执行 INTST6 中断服务之前，可能已经完成下一次发送。作为一个应对措施，开发一个程序，通过访问 TXSF6 标志来计数发送数据的数量，可以执行检测。

图 15-18 是连续发送处理流程的示例。

图 15-18. 连续发送处理流程的示例

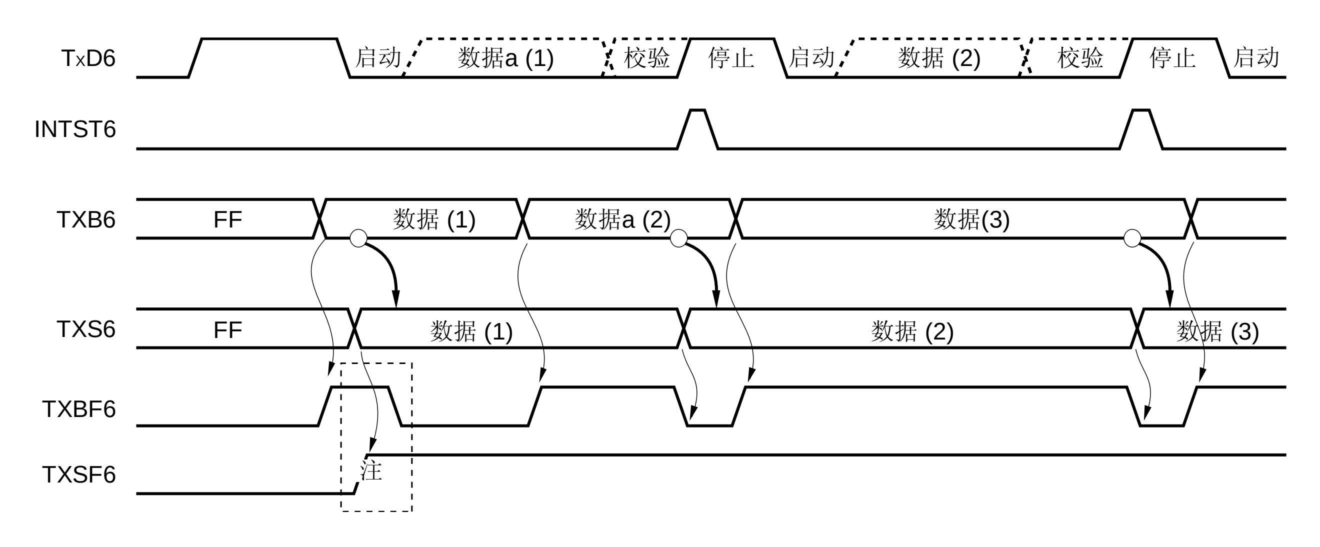
设置寄存器
写入 TXB6
已执行了必须的发送次数？
是
否
读取 ASIF6 TXBF6 = 0?
否
是
写入 TXB6
产生发送完成中断？
否
是
已执行了必须的发送次数？
是
否
读取ASIF6 TXSF6 = 0?
否
是
发送过程完成

备注

TXB6：	发送缓冲寄存器 6
ASIF6：	异步串行接口发送状态寄存器 6
TXBF6：	ASIF6 的第 1 位（发送缓冲器数据标志位）
TXSF6：	ASIF6 的第 0 位（发送移位寄存器数据标志位）

图 15-19 显示了启动连续发送的时序，而图 15-20 显示了结束连续发送的时序。

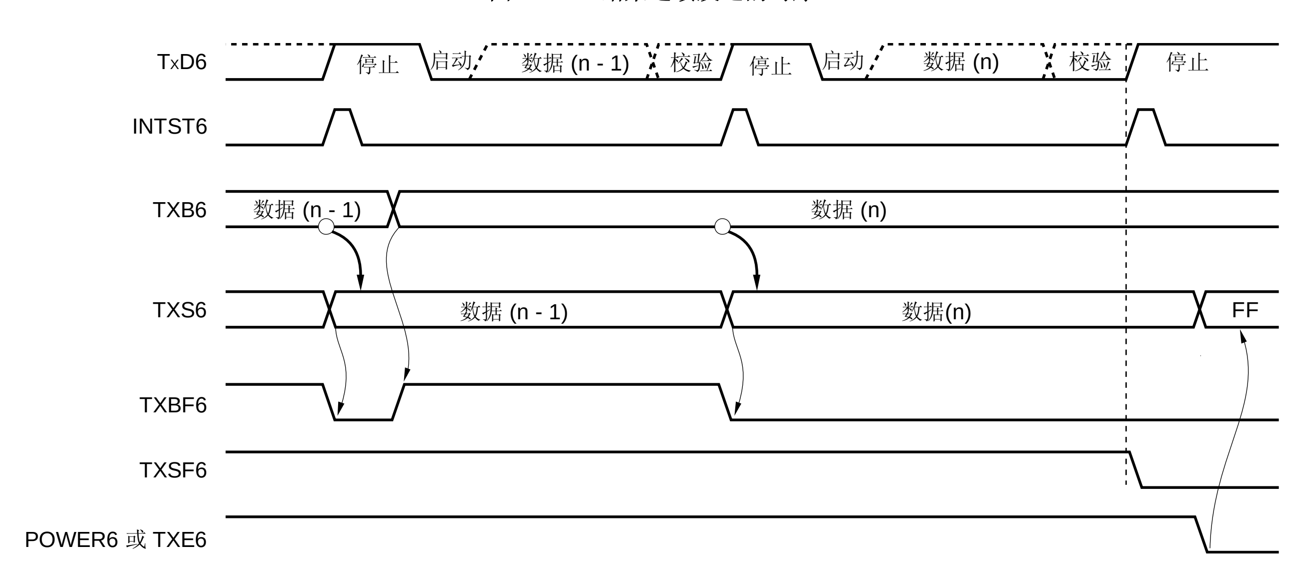
图 15-19. 启动连续发送的时序



注　　在读取 ASIF6 时，有一个时期 TXBF6 和 TXSF6 = 1，1。因此，仅使用 TXBF6 就可以判断是否允许写入。

备注　TxD6：　TxD6 引脚 （输出）
　　　INTST6：　中断请求信号
　　　TXB6：　发送缓冲寄存器 6
　　　TXS6：　发送移位寄存器 6
　　　ASIF6：　异步串行接口发送状态寄存器 6
　　　TXBF6：　ASIF6 的第 1 位
　　　TXSF6：　ASIF6 的第 0 位

图 15-20. 结束连续发送的时序



备注

TxD6：	TxD6 引脚 （输出）
INTST6：	中断请求信号
TXB6：	发送缓冲寄存器 6
TXS6：	发送移位寄存器 6
ASIF6：	异步串行接口发送状态寄存器 6
TXBF6：	ASIF6 的第 1 位
TXSF6：	ASIF6 第 0 位
POWER6：	异步串行接口操作模式寄存器 （ASIM6）的第 7 位
TXE6：	异步串行接口操作模式寄存器 （ASIM6）的第 6 位

（e） 正常接收

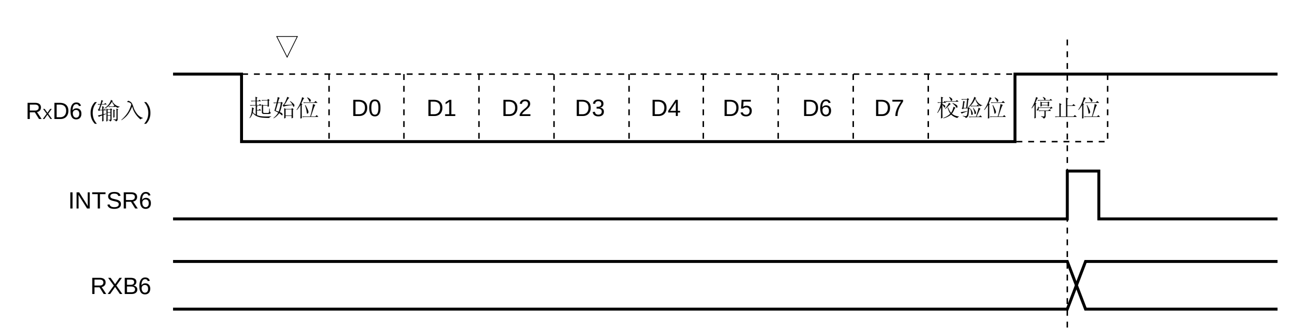
当异步串行接口操作模式寄存器 6（ASIM6）的第 7 位（POWER6）被设置为 1，然后将 ASIM6 的第 5 位（RXE6）设置为 1 时，使能接收，并对 RxD6 引脚输入进行采样。

当检测到 RxD6 引脚输入的下降沿时，波特率发生器的 8 位计数器开始计数。当计数值达到波特率发生器控制寄存器 6（BRGC6）的设置值时，再次采样 RxD6 引脚输入 （如图 15-21 中的 ▽ ）。如果此时 RxD6 为低电平，则输入信号被认为是起始位。

当检测到起始位时，开始接收，按照设置的波特率将串行数据顺序存入接收移位寄存器 6（RXS6）。当接收到停止位时，会产生接收完成中断信号（INTSR6），并且 RXS6 的数据被写入接收缓冲寄存器 6（RXB6）。但是，如果发生溢出错误（OVE6），则接收数据不写入 RXB6。

即使在接收进行时发生校验错（PE6），接收仍然继续，直至接收到停止位，而在接收完成后，会产生一个接收错误中断（INTSR6/INTSRE6）。

图 15-21. 接收完成中断请求时序



注意事项 **1.** 如果发生接收错误，先读取 **ASIS6**，然后再读取 **RXB6**，可以清除错误标志。否则，当接收到下一个数据时会产生溢出错误，且接收错误状态保持不变。

2. 总是按照 **“停止位的数量 = 1”**来执行接收，第 **2** 个停止位被忽略。

3. 在读取 **RXB6** 之前，请确保先读取异步串行接口接收错误状态寄存器 **6**（**ASIS6**）。

（f） 接收错误

在接收期间可能发生三种类型的错误：校验错误、帧错误或溢出错误。作为数据接收的结果，如果异步串行接口接收错误状态寄存器 6（ASIS6）的错误标志位被设置，则将会产生一个接收错误中断请求信号（INTSR6/INTSRE6）。

在接收错误中断服务（INTSR6/INTSRE6）中读取 ASIS6 的内容，可以识别接收期间发生了哪种错误（参见**图 15-6**）。

读取 ASIS6 后，ASIS6 的内容被清除为 0。

表 15-3. 产生接收错误的原因

接收错误	产生原因
校验错误	发送数据的校验位与接收到的数据校验位不匹配
帧错误	未检测到停止位
溢出错误	在从接收缓冲寄存器 0（RXB0）读取数据之前，已经完成下一个数据的接收

通过将异步串行接口操作模式寄存器 6（ASIM6）的第 0 位（ISRM6）清零，可以把接收错误中断划分为接收完成中断（INTSR6）和错误中断（INTSRE6）两部分。

图 15-22. 接收错误中断

1. 如果 ISRM6 被清除为 0（接收完成中断（INTSR6）和错误中断（INTSRE6）是分离的）

（**a**） 接收期间无错误

INTSR6

INTSRE6

（**b**） 接收期间产生错误

INTSR6

INTSRE6

2. 如果 ISRM6 被设置为 1（接收错误中断包含在 INTSR6 中）

（**a**） 接收期间无错误

INTSR6

INTSRE6

（**b**） 接收期间产生错误

INTSR6

INTSRE6

（g） 接收数据的噪声过滤器

通过预分频器设置的基准时钟输出，采样 RxD0 信号。

如果两次采样值相同，则匹配检测器的输出会改变，且被采样的数据作为输入数据。

噪音过滤器的电路结构如图 15-23 所示，外部信号状态之后，延迟两个时钟执行接收操作的内部处理。

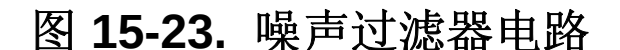
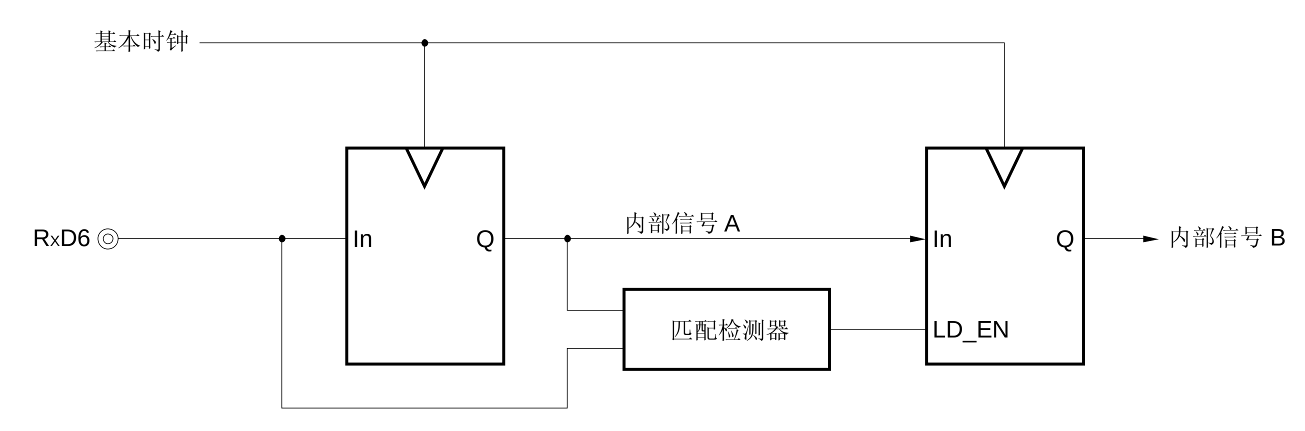


图 15-23. 噪声过滤器电路



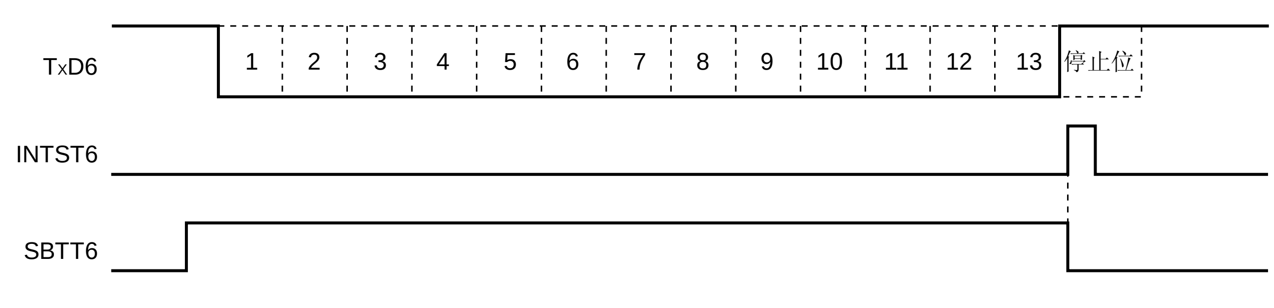
（h） SBF 发送

该设备用于 LIN 通信操作时，可使用 SBF（同步间隔场）发送控制功能进行发送。LIN 的发送操作参见**图 15-1 LIN 发送操作**。

当异步串行接口模式寄存器 6（ASIM6）的第 7 位（POWER6）被设置为 1 时，TxD6 引脚输出高电平。接下来当 ASIM6 的第 6 位（TXE6）被设置为 1 时，进入发送使能状态。且通过将异步串行接口控制寄存器 6（ASICL6）的第 5 位（SBTT6）设置为 1，可以启动 SBF 发送。

在此之后，一个 13 位至 20 位（由 ASICL6 的第 4 位至第 2 位（SBL62 至 SBL60）设置）的低电平被输出。SBF 发送结束后，产生发送完成中断请求（INTST6），且 SBTT6 被自动清除。此后，恢复到正常发送模式。发送被停止，直到待发送数据下一次被写入缓冲寄存器 6（TXB6）或直到 SBTT6 被设置为 1。

图 15-24. SBF 发送



备注 TxD6： TxD6 引脚（输出）
INTST6： 发送完成中断请求
SBTT6： 异步串行接口控制寄存器 6（ASICL6）的第 5 位

（i） **SBF 接收**

该设备用于 LIN 通信操作时，SBF（同步间隔场）接收控制功能用于接收操作。LIN 的接收操作参见图 **15-2 LIN 接收操作**。

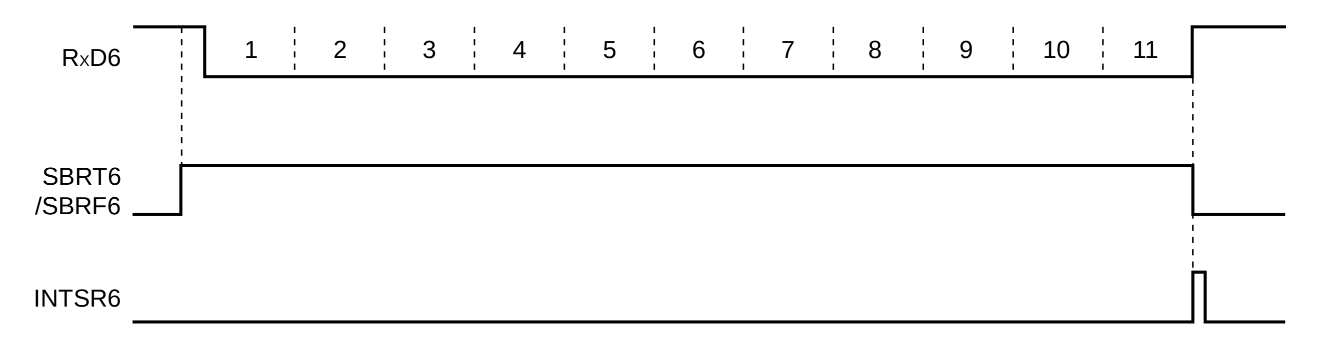
当异步串行接口模式寄存器 6（ASIM6）的第 7 位（POWER6）被设置为 1，然后再将 ASIM6 的第 5 位（RXE6）设置为 1 时，接收被使能。当异步串行接口控制寄存器 6（ASICL6）的第 6 位（SBRT6）被设置为 1 时，允许 SBF 接收。在 SBF 接收使能状态下，和正常接收使能状态的方式一样，对 RxD6 引脚进行采样，并检测起始位。

当检测到起始位时，启动接收，按照设置的波特率将串行数据顺序存入接收移位寄存器 6（RXS6）。当接收到停止位时，且 SBF 的宽度大于或等于 11 位时，与正常处理一样，产生接收完成中断请求（INTSR6）。此时 SBRF6 和 SBRT6 被自动清除，SBF 接收结束。不再检测错误，比如 OVE6、PE6 和 FE6（异步串行接口接收错误状态寄存器 6（ASIS6）的第 0 位至第 2 位），且不执行 UART 通信的错误检测处理。

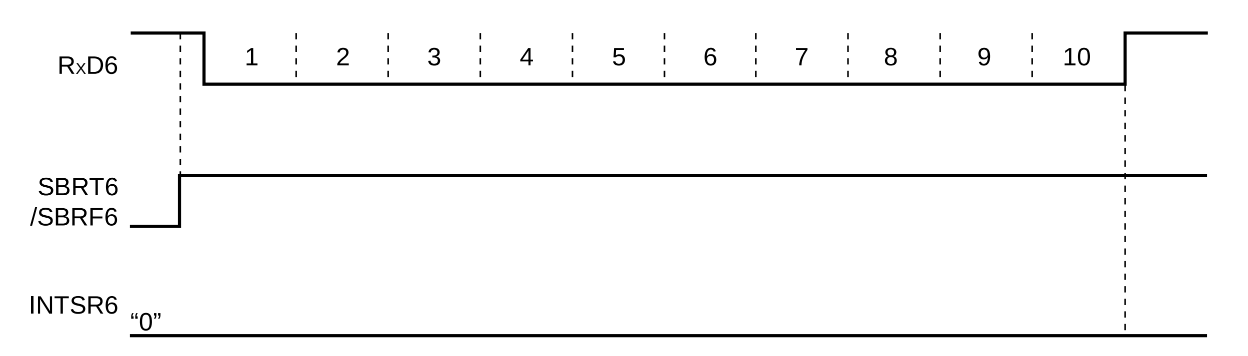
此外，不执行接收移位寄存器 6（RXS6）和接收缓冲寄存器 6（RXB6）之间的数据传送，并保留复位值 FFH。如果 SBF 的宽度小于或等于 10 位，则在接收到停止位后不会发生中断也没有错误处理，并恢复 SBF 接收模式。这种情况下，不清除 SBRF6 和 SBRT6 位。

图 15-25. SBF 接收

1. 正常 **SBF** 接收（检测到停止位，且 **SBF** 宽度大于 **10.5** 位）



2. SBF 接收错误（检测到停止位，且 **SBF** 宽度小于或等于 **10.5** 位）



备注
RxD6： RxD6 引脚（输入）
SBRT6： 异步串行接口控制寄存器 6 （ASICL6） 的第 6 位
SBRF6： ASICL6 的第 7 位
INTSR6： 接收完成中断请求

15.4.3 专用波特率发生器

专用波特率发生器由一个源时钟选择器和一个 8 位可编程计数器组成，用于为 UART0 发送/接收产生一个串行时钟。

为发送和接收提供不同的 8 位计数器。

（1） 波特率发生器的配置

- 基准时钟

  该时钟由时钟选择寄存器 6（CKSR6）的第 3 位至第 0 位（TPS63 至 TPS60）选择。当异步串行接口操作模式寄存器 0（ASIM0）的第 7 位（POWER6）为 1 时，该时钟提供给各个模块。该时钟称为基准时钟，它的频率被称为 f_{XCLK6}。当 POWER6 = 0 时，基准时钟被固定为低电平。

- 发送计数器

  当异步串行接口操作模式寄存器 6（ASIM6）的第 7 位（POWER6）或第 6 位（TXE6）为 0 时，该计数器停止计数，并被清除为 0。

  当 POWER6 = 1 且 TXE6 = 1 时，计数器开始计数。

  当第 1 个待发送数据被写入发送移位寄存器 6（TXS6）时，计数器被清除为 0。

  如果数据被连续发送，则当 1 帧数据发送完成时，计数器再次被清除为 0。如果后面没有数据要发送，则计数器不被清除为 0，而是继续计数直至 POWER6 或 TXE6 被清除为 0。

- 接收计数器

  当异步串行接口操作模式寄存器 6（ASIM6）的第 7 位（POWER6）或第 5 位（RXE6）为 0 时，该计数器停止计数，并被清除为 0。

  当检测到起始位时，计数器开始计数。

  接收到一帧数据后，计数器停止计数，直到检测到下一个起始位。

图 **15-26.** 波特率发生器的配置

POWER6

f_{PRS}
$f_{PRS}/2$
$f_{PRS}/2^2$
$f_{PRS}/2^3$
$f_{PRS}/2^4$
$f_{PRS}/2^5$
$f_{PRS}/2^6$
$f_{PRS}/2^7$
$f_{PRS}/2^8$
$f_{PRS}/2^9$
$f_{PRS}/2^{10}$
8位定时器/
事件计数器
50 输出

选择器

f_{XCLK6}

波特率发生器

POWER6，TXE6 (或 RXE6)

8位计数器

匹配检测器

1/2

波特率

CKSR6：TPS63 至 TPS60

BRGC6：MDL67 至 MDL60

备注 POWER6： 异步串行接口操作模式寄存器 6 （ASIM6）的第 7 位
TXE6： ASIM6 的第 6 位
RXE6： ASIM6 的第 5 位
CKSR6： 时钟选择寄存器 6
BRGC6： 波特率发生器控制寄存器 6

（2） 串行时钟的产生

使用波特率发生器控制寄存器 0（BRGC0）可产生指定将要产生的串行时钟。
使用 BRGC0 的第 7 位和第 6 位（TPS01 和 TPS00），可以选择输入到 5 位计数器的时钟。
BRGC0 的第 4 位至第 0 位（MDL04 至 MDL00）可以用来选择 5 位计数器的分频值。

使用时钟选择寄存器 6（CKSR6）和波特率发生器控制寄存器 6 （BRGC6）可产生指定的串行时钟。
使用 CKSR6 的第 3 位至第 0 位（TPS63 至 TPS60），可以设置输入到 8 位计数器的时钟。使用 BRGC6 的第 7 位至 第 0 位（MDL67 至 MDL60）可以设置 8 位计数器的分频值（$f_{XCLK6}/4$ 至 $f_{XCLK6}/255$）。

15.4.4 波特率的计算

（1） 波特率计算表达式

波特率可由下列表达式计算。

- 波特率 = $\frac{f_{XCLK6}}{2 \times k}$ [bps]

f_{XCLK6}： 由 CKSR6 的 TPS63 至 TPS60 位选择的基准时钟的频率

k： 由 BRGC6 的 MDL67 至 MDL60 位设置的值（k = 4，5，6，...，255）

表 15-4. TPS63 至 TPS60 的设定值

TPS63	TPS62	TPS61	TPS60	基准时钟 （f_{XCLK6}） 选择注 1				
					f_{PRS} = 2 MHz	f_{PRS} = 5 MHz	f_{PRS} = 8 MHz	f_{PRS} = 10 MHz
0	0	0	0	f_{PRS} 注 2	2 MHz	5 MHz	8 MHz	10 MHz
0	0	0	1	$f_{PRS}/2$	1 MHz	2.5 MHz	4 MHz	5 MHz
0	0	1	0	$f_{PRS}/2^2$	500 kHz	1.25 MHz	2 MHz	2.5 MHz
0	0	1	1	$f_{PRS}/2^3$	250 kHz	625 kHz	1 MHz	1.25 MHz
0	1	0	0	$f_{PRS}/2^4$	125 kHz	312.5 kHz	500 kHz	625 kHz
0	1	0	1	$f_{PRS}/2^5$	62.5 kHz	156.25 kHz	250 kHz	312.5 kHz
0	1	1	0	$f_{PRS}/2^6$	31.25 kHz	78.13 kHz	125 kHz	156.25 kHz
0	1	1	1	$f_{PRS}/2^7$	15.625 kHz	39.06 kHz	62.5 kHz	78.13 kHz
1	0	0	0	$f_{PRS}/2^8$	7.813 kHz	19.53 kHz	31.25 kHz	39.06 kHz
1	0	0	1	$f_{PRS}/2^9$	3.906 kHz	9.77 kHz	15.625 kHz	19.53 kHz
1	0	1	0	$f_{PRS}/2^{10}$	1.953 kHz	4.88 kHz	7.813 kHz	9.77 kHz
1	0	1	1	TM50 输出注 3				
其它情况				禁止设置				

注

1. 当外设硬件时钟 （f_{PRS}）使用高速系统时钟（f_{XH}） （XSEL = 1），根据供电电压的不同，f_{PRS} 的工作频率也不同。

- V_{DD} = 2.7 至 5.5 V： $f_{PRS} \leq 10$ MHz
- V_{DD} = 1.8 至 2.7 V： $f_{PRS} \leq 5$ MHz

2. 如果外设硬件时钟 （f_{PRS}）使用内部高速振荡时钟（f_{RH}） （XSEL = 0），当 1.8 V ≤ V_{DD} < 2.7 V 时，禁止设置 TPS63 = TPS62 = TPS61 = TPS60 = 0 （基时钟： f_{PRS}）。

3. 当选择 TM50 输出作为基准时钟时，应注意以下几点。

- 在 TM50 与 CR50 匹配（TMC506 = 0）时计数时钟被清除并启动的模式
  先启动 8 位定时器/事件计数器 50，然后使能定时器 F/F 反转操作（TMC501 = 1）。
- PWM 模式 （TMC506 = 1）
  先启动 8 位定时器/事件计数器 50，然后设置计数时钟，使占空比 = 50%。

不论何种模式，都无需使能（TOE50 = 1）TO50 输出。

（2） 波特率误差

通过下列表达式计算波特率误差。

$$\bullet\ 误差（\%）=\left(\frac{实际波特率\ (有误差的波特率)}{预期波特率\ (正确的波特率)}-1\right)\times 100\ [\%]$$

注意事项 1. 在发送期间，必须保持波特率误差在接收目的方允许误差范围内。

2. 在接收期间，波特率的误差必须满足“（4）接收期间允许的波特率范围”的范围。

举例： 基准时钟频率 = 10 MHz = 10,000,000 Hz
BRGC6 的 MDL67 至 MDL60 位的设置值 = 00100001B （k = 33）
目标波特率 = 153600 bps

波特率 = 10 M / （2 × 33）
= 10000000 / （2 × 33） = 151,515 [bps]

误差 = （151515/153600 – 1） × 100
= –1.357 [%]

（3） 波特率设置举例

表 15-5. 波特率发生器的设置数据

波特率 [bps]	f_{PRS} = 2.0 MHz				f_{PRS} = 5.0 MHz				f_{PRS} = 10.0 MHz			
	TPS63-TPS60	k	计算值	ERR [%]	TPS63-TPS60	k	计算值	ERR [%]	TPS63-TPS60	k	计算值	ERR [%]
300	8H	13	301	0.16	7H	65	301	0.16	8H	65	301	0.16
600	7H	13	601	0.16	6H	65	601	0.16	7H	65	601	0.16
1200	6H	13	1202	0.16	5H	65	1202	0.16	6H	65	1202	0.16
2400	5H	13	2404	0.16	4H	65	2404	0.16	5H	65	2404	0.16
4800	4H	13	4808	0.16	3H	65	4808	0.16	4H	65	4808	0.16
9600	3H	13	9615	0.16	2H	65	9615	0.16	3H	65	9615	0.16
19200	2H	13	19231	0.16	1H	65	19231	0.16	2H	65	19231	0.16
24000	1H	21	23810	–0.79	3H	13	24038	0.16	4H	13	24038	0.16
31250	1H	16	31250	0	4H	5	31250	0	5H	5	31250	0
38400	1H	13	38462	0.16	0H	65	38462	0.16	1H	65	38462	0.16
48000	0H	21	47619	–0.79	2H	13	48077	0.16	3H	13	48077	0.16
76800	0H	13	76923	0.16	0H	33	75758	–1.36	0H	65	76923	0.16
115200	0H	9	111111	–3.55	1H	11	113636	–1.36	0H	43	116279	0.94
153600	–	–	–	–	1H	8	156250	1.73	0H	33	151515	–1.36
312500	–	–	–	–	0H	8	312500	0	1H	8	312500	0
625000	–	–	–	–	0H	4	625000	0	1H	4	625000	0

备注 TPS63 至 TPS60：时钟选择寄存器 6 （CKSR6）的第 3 位至第 0 位（基准时钟（f_{XCLK6}）的设置）

k： 使用波特率发生器控制寄存器 6 （BRGC6）的 MDL67 至 MDL60 位设置的值（k = 4，5，6，...，255）

f_{PRS}： 外设硬件时钟频率

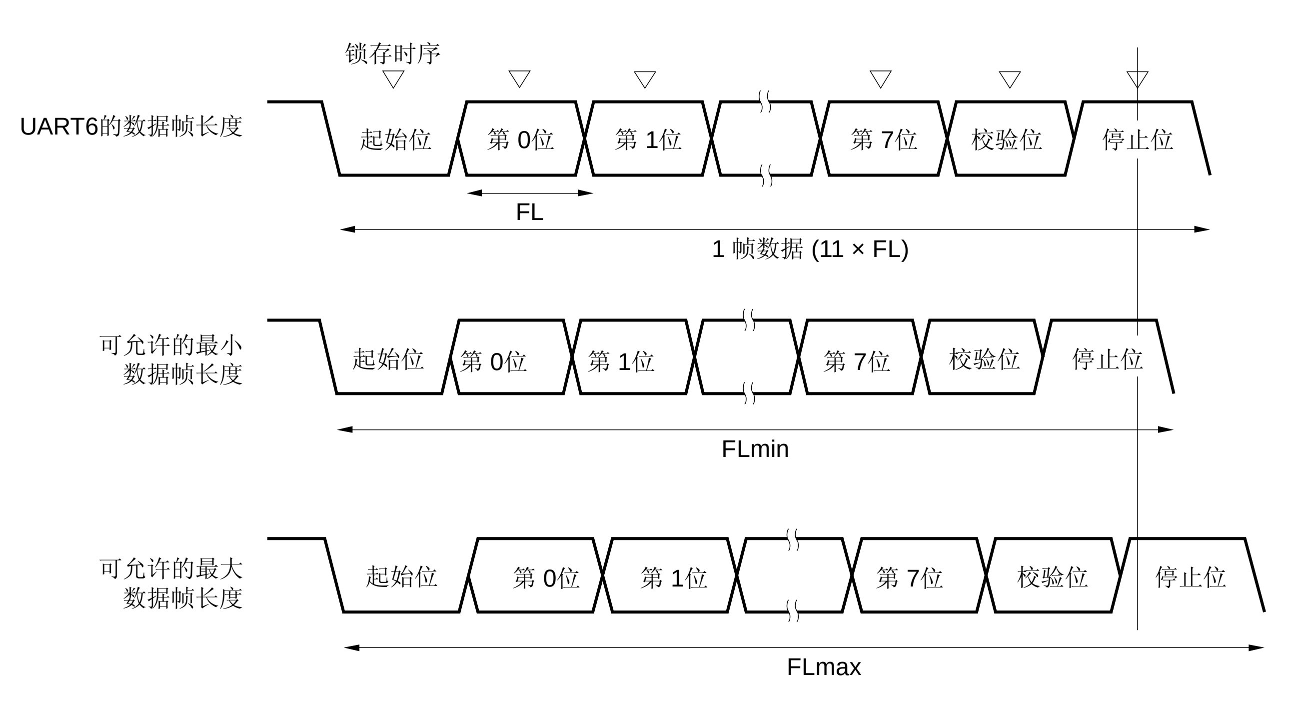
ERR： 波特率误差

（4） 接收期间允许的波特率范围

接收期间，发送目的方的波特率允许误差范围如下所示。

注意事项 接收期间，必须确保波特率误差在允许的误差范围内，可以使用以下表达式计算。

图 15-27. 接收期间允许的波特率范围



如图 15-27 所示，当检测到起始位后，接收数据的锁存时序由波特率发生器控制寄存器 6（BRGC6）设置的计数器确定。如果最后的数据（停止位）满足该锁存时序，则数据可被正确接收。

假定接收数据为 11 位，各项理论值计算如下。

$FL = (Brate)^{-1}$

Brate： UART6 的波特率
k： BRGC6 的设置值
FL： 1 位数据长度
锁存时序的余量： 2 个时钟

可允许的最小数据帧长度：
$$FLmin = 11 \times FL - \frac{k-2}{2k} \times FL = \frac{21k+2}{2k} FL$$

因此，发送目的方可以接收的最大波特率如下所示。

$$BRmax = (FLmin/11)^{-1} = \frac{22k}{21k+2} Brate$$

同样，允许的最大数据帧长度可以计算如下。

$$\frac{10}{11} \times FLmax = 11 \times FL - \frac{k+2}{2 \times k} \times FL = \frac{21k-2}{2 \times k} FL$$

$$FLmax = \frac{21k-2}{20k} FL \times 11$$

因此，发送目的方可以接收的最小波特率如下所示。

$$BRmin = (FLmax/11)^{-1} = \frac{20k}{21k-2} Brate$$

通过上述最小和最大波特率表达式，可以计算 UART6 与发送目的方之间允许的波特率误差，如下所示。

表 15-6. 允许的最大/最小波特率误差

分频比（k）	允许的最大波特率误差	允许的最小波特率误差
4	+2.33%	–2.44%
8	+3.53%	–3.61%
20	+4.26%	–4.31%
50	+4.56%	–4.58%
100	+4.66%	–4.67%
255	+4.72%	–4.73%

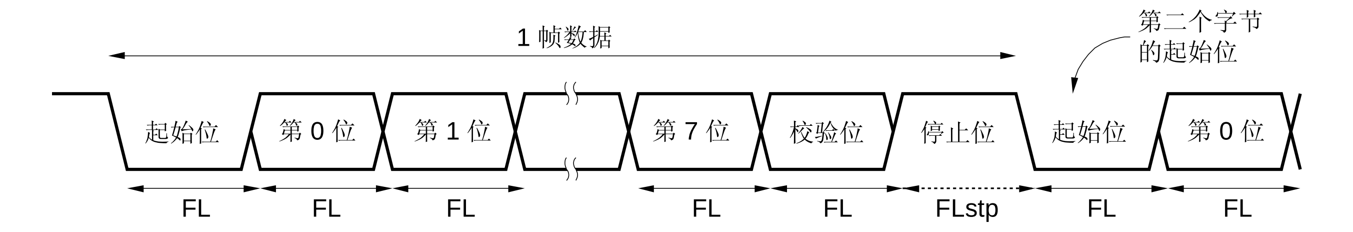
备注 **1.** 接收的允许误差取决于一帧的位数量、输入时钟频率和分频比（k）。输入时钟频率越高，且分频比（k）越高，允许的误差就越大。

2. k： BRGC6 的设置值。

（5） 连续发送期间数据帧的长度

当连续发送数据时，从停止位到下一个起始位的数据帧长度比正常值延长了两个基准时钟。但是，因为在接收端检测到起始位时对时序进行初始化，因此通信结果不会受到影响。

图 15-28. 连续发送期间数据帧长度



当 1 位数据长度为 FL、停止位长度为 FLstp，基准时钟频率为 f_{XCLK6} 时，有以下表达式成立。

$$FLstp = FL + 2/f_{XCLK6}$$

因此，连续发送期间的数据帧长度为：

$$数据帧长度 = 11 \times FL + 2/f_{XCLK6}$$

16.1 串行接口CSI10 的功能

串行接口 CSI10 具有以下两种模式。

（1） 操作停止模式

不执行串行通信时可以使用该模式，可以降低功耗。

详情参见 **16.4.1 操作停止模式**。

（2） 3 线串行 I/O 模式（可选择 MSB / LSB-先行）

该模式以 8 位单元进行连续数据通信，采用三条线：串行时钟线（$\overline{SCK10}$）和串行数据线（SI10 和 SO10））。

在 3 线串行 I/O 模式下，数据通信的处理时间可以缩短，因为发送和接收操作可以同时执行。

此外，可以指定 8 位数据通信是 MSB 或者 LSB-先行，于是该接口可以和任何设备连接。

详情参见 **16.4.2 3 线串行 I/O 模式**。

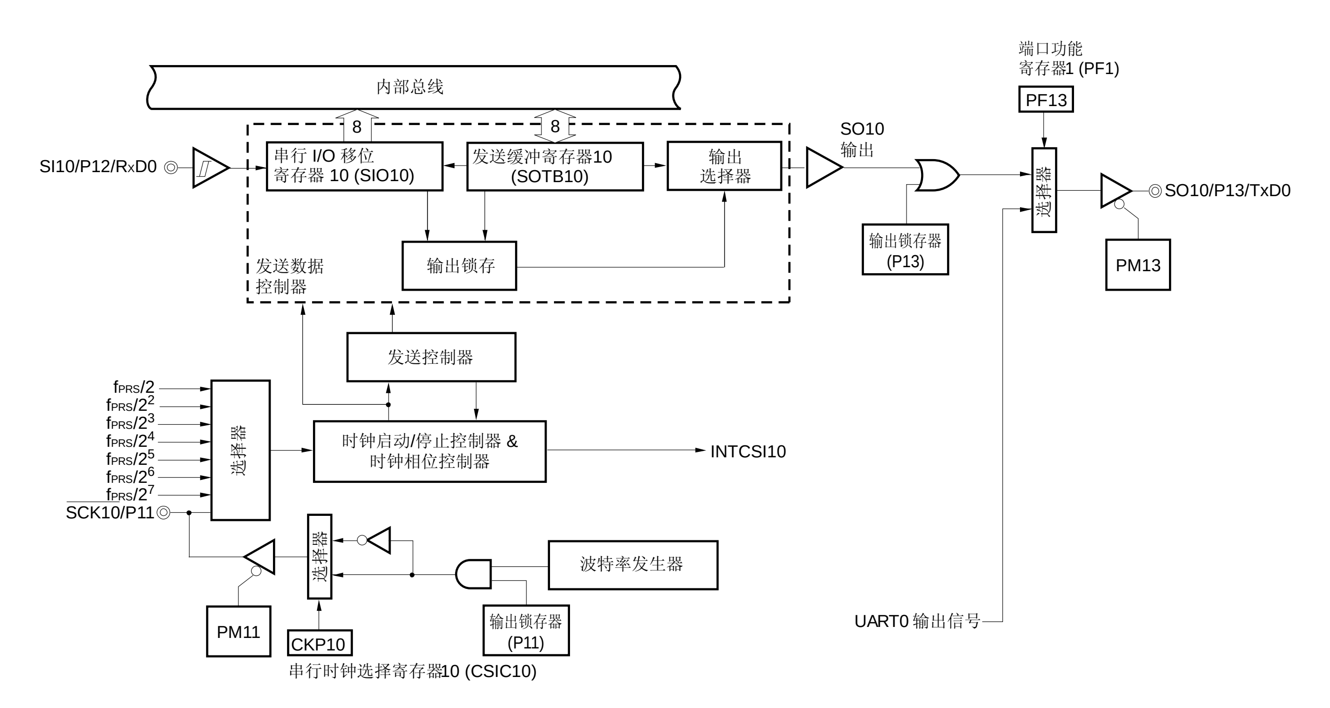
16.2 串行接口CSI10 的配置

串行接口 CSI10 由以下硬件组成。

表 16-1 串行接口 CSI10 的配置

项目	配置
控制器	发送控制器 时钟启动/停止控制器&时钟相位控制器
寄存器	发送缓冲寄存器 10（SOTB10） 串行 I/O 移位寄存器 10（SIO10）
控制寄存器	串行操作模式规范寄存器 10（CSIM10） 串行状态寄存器 10（CSIC10） 端口功能寄存器 1（PF1） 端口模式寄存器 1（PM1） 端口寄存器 1（P1）

图 16-1. 串行接口 CSI10 的框图



（1） 发送缓冲寄存器 10（SOTB10）

该寄存器用于设置发送数据。

当串行操作模式寄存器 10（CSIM10）的第 7 位（CSIE10）和第 6 位（TRMD10）均为 1 时，将数据写入 SOTB10，可以启动发送/接收操作。

使用串行 I/O 移位寄存器 10 将写入 SOTB10 的并行数据转换成串行数据，并把数据输出到串行输出引脚（SO10）。

可以通过 8 位存储器操作指令对 SOTB10 进行读写操作。

复位信号的产生会将该寄存器清除为 00H。

注意事项　　**当 CSOT10 = 1**（在串行通信期间）时，不要访问 **SOTB10**。

（2） 串行 I/O 移位寄存器 10 （SIO10）

该 8 位寄存器可以将并行数据转换成串行数据，或将串行数据转换成并行数据。

可以通过 8 位存储器操作指令对该寄存器进行读取。

如果串行操作模式寄存器 10 （CSIM10）的第 6 位（TRMD10）为 0，则从 SIO10 读取数据，可以启动接收。

接收期间，将数据从串行输入引脚（SI10）读到 SIO10 中。

复位信号的产生会将该寄存器清除为 00H。

注意事项　　**当 CSOT10 = 1**（在串行通信期间）时，不要访问 **SIO10**。

16.3 控制串行接口CSI10 的寄存器

串行接口 CSI10 由以下五个寄存器控制。

- 串行操作模式寄存器 10 （CSIM10）
- 串行时钟选择寄存器 10 （CSIC10）
- 端口功能寄存器 1 （PF1）
- 端口模式寄存器 1 （PM1）
- 端口寄存器 1 （P1）

（1） 串行操作模式寄存器 10 （CSIM10）

CSIM10 用于选择操作模式、使能或禁止操作。

可以通过 8 位存储器操作指令来设置 CSIM10。

复位信号的产生会将该寄存器清除为 00H。

图 16-2. 串行操作模式寄存器 10（CSIM10）的格式

地址： FF80H 复位后： 00H R/W 注 1

符号	<7>	6	5	4	3	2	1	0
CSIM10	CSIE10	TRMD10	0	DIR10	0	0	0	CSOT10

CSIE10 注 2	3 线串行 I/O 模式中的操作控制
0	禁止操作且异步复位内部电路注 3
1	使能操作

TRMD10 注 4	发送/接收模式控制
0 注 5	接收模式（禁止发送）
1	发送/接收模式

DIR10 注 6	首位规范
0	MSB
1	LSB

CSOT10	通信状态标志
0	通信停止
1	通信正在进行中

注 **1.** 第 0 位只读。

2. 要将 P11/$\overline{\text{SCK10}}$, P12/SI10/RxD0/<RxD6>和 P13/SO10/TxD0/<TxD6>用作通用端口，清除 CSIE10 为 0。

3. CSIM10 的第 0 位（CSOT10）和串行 I/O 移位寄存器 10 （SIO10）被重置。

4. 当 CSOT10 = 1（在串行通信期间）时，不要重写 TRMD10。

5. 当 TRMD10 为 0 时，SO10 输出（参见图 **16-1**）固定为低电平。当从 SIO10 读取数据时，启动接收。

6. 当 CSOT10 = 1（在串行通信期间）时，不要重写 DIR10。

注意事项 1. 当从待机模式中恢复操作时，请确保在将中断请求标志寄存器 0H（IF0H）的第 2 位清除（0）之后。

2. 请确保将第 5 位清除为 0。

（2） 串行时钟选择寄存器 10 （CSIC10）

该寄存器指定数据发送/接收的时序，并设置串行时钟。

可以通过 8 位存储器操作指令来设置 CSIC10。

复位信号的产生会将该寄存器清除为 00H。

图 16-3. 串行时钟选择寄存器 10（CSIC10）的格式

地址： FF81H 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
CSIC10	0	0	0	CKP10	DAP10	CKS102	CKS101	CKS100

CKP10	DAP10	数据发送/接收时序的规范	类型
0	0	$\overline{SCK10}$ SO10 D7 D6 D5 D4 D3 D2 D1 D0 SI10 输入时序	1
0	1	$\overline{SCK10}$ SO10 D7 D6 D5 D4 D3 D2 D1 D0 SI10 输入时序	2
1	0	$\overline{SCK10}$ SO10 D7 D6 D5 D4 D3 D2 D1 D0 SI10 输入时序	3
1	1	$\overline{SCK10}$ SO10 D7 D6 D5 D4 D3 D2 D1 D0 SI10 输入时序	4

CKS102	CKS101	CKS100	CSI10 串行时钟的选择[注 1, 2]					模式
				f_{PRS} = 2 MHz	f_{PRS} = 5 MHz	f_{PRS} = 8 MHz	f_{PRS} = 10 MHz	
0	0	0	$f_{PRS}/2$	1 MHz	2.5 MHz	4 MHz	禁止设置	主设备模式
0	0	1	$f_{PRS}/2^2$	500 kHz	1.25 MHz	2 MHz	2.5 MHz	
0	1	0	$f_{PRS}/2^3$	250 kHz	625 kHz	1 MHz	1.25 MHz	
0	1	1	$f_{PRS}/2^4$	125 kHz	312.5 kHz	500 kHz	625 kHz	
1	0	0	$f_{PRS}/2^5$	62.5 kHz	156.25 kHz	250 kHz	312.5 kHz	
1	0	1	$f_{PRS}/2^6$	31.25 kHz	78.13 kHz	125 kHz	156.25 kHz	
1	1	0	$f_{PRS}/2^7$	15.63 kHz	39.06 kHz	62.5 kHz	78.13 kHz	
1	1	1	输入到 $\overline{SCK10}$ 的外部时钟					从设备模式

注　**1.** 如果外设硬件时钟（f_{PRS}）使用高速系统时钟（f_{XH}）（XSEL ＝ 1），根据供电电压的不同，f_{PRS} 的工作频率也不同。

- V_{DD} = 2.7 至 5.5 V： $f_{PRS} \leq 10$ MHz
- V_{DD} = 1.8 至 2.7 V： $f_{PRS} \leq 5$ MHz

注　　**2.**　串行时钟的设置需要满足下列条件。

- V_{DD} = 2.7 至 5.5 V：　串行时钟 ≤ 4 MHz
- V_{DD} = 1.8 至 2.7 V：　串行时钟 ≤ 2 MHz

注意事项　1.　当 CSIE10=1 时（使能操作），不要向 CSIC10 写入。

2.　要将 P11/$\overline{\text{SCK10}}$ 和 P13/SO10/TxD0/<TxD6>用作通用端口，设置 CSIC10 为默认状态（00H）。

3.　复位后数据时钟的相位类型是类型 1。

备注　　f_{PRS}：外设硬件时钟频率

（3）　端口功能寄存器 1 （PF1）

该寄存器用于设置 P13/SO10/TxD0/<TxD6>的引脚功能。

可以通过 8 位存储器操作指令来设置 PF1。

复位信号的产生会将 PF1 清除为 00H。

图 16-4.　端口功能寄存器 1 （PF1）的格式

地址：FF20H　复位后：00H　R/W

符号	7	6	5	4	3	2	1	0
PF1	0	0	0	0	PF13	0	0	0

PF13	端口（P13），CSI10、UART0和 UART06输出选择
0	用作P13或 SO10
1	用作TxD0或TxD6

（4） 端口模式寄存器 1 （PM1）

该寄存器用于按位设置端口 1 的输入/输出模式。

当 P11/$\overline{SCK10}$ 引脚用于串行接口时钟输出时，将 PM11 清除为 0，设置 P11 输出锁存器为 1。

当 P13/SO10/TxD0/<TxD6>引脚用于串行接口数据输出时，将 PM13 清除为 0 且设置 P13 输出锁存器为 0。

当 P11/$\overline{SCK10}$ 作为串行接口的时钟输入引脚，且 P12/SI10/RxD0/<RxD6>作为数据输入引脚时，设置 PM11 和 PM12 为 1。此时， P11 和 P12 的输出锁存值可能是 0 或 1。

可以通过 1 位或 8 位存储器操作指令来设置 PM1。

复位信号的产生会将该寄存器设置为 FFH。

图 16-5.端口模式寄存器 1 （PM1）的格式

地址：FF21H　　复位后：FFH　　R/W

符号	7	6	5	4	3	2	1	0
PM1	PM17	PM16	PM15	PM14	PM13	PM12	PM11	PM10

PM1n	P1n 引脚 I/O 模式选择 (n = 0 至 7)
0	输出模式 (输出缓冲器打开)
1	输入模式 (输出缓冲器关闭)

16.4 串行接口CSI10 的操作

串行接口 CSI10 可在以下两种模式中使用。

- 操作停止模式
- 3 线串行 I/O 模式

16.4.1 操作停止模式

该模式下，不执行串行通信，因此，可以降低功耗。此外，在该模式下 P11/$\overline{\text{SCK10}}$，P12/SI10/RxD0 和 P13/SO10/TxD0 引脚可用作普通 I/O 端口引脚。

（1） 使用的寄存器

由串行操作模式寄存器 10（CSIM10）设置操作停止模式。

通过将 CSIM10 的第 7 位（CSIE10）清除为 0，设置操作停止模式。

（a） 串行操作模式寄存器 10 （CSIM10）

可以通过 1 位或 8 位存储器操作指令来设置 CSIM10。

复位信号的产生会将 CSIM10 清除为 00H。

地址： FF80H 复位后： 00H R/W

符号	<7>	6	5	4	3	2	1	0
CSIM10	CSIE10	TRMD10	0	DIR10	0	0	0	CSOT10

CSIE10	3 线串行 I/O 模式中的操作控制
0	禁止操作[注1]，并异步复位内部电路[注2]

注 **1.** 要将 P11/$\overline{\text{SCK10}}$，P12/SI10/RxD0/<RxD6>和 P13/SO10/TxD0/<TxD6>用作通用端口，设置 CSIM10 为默认状态（00H）。

2. CSIM10 的第 0 位 （CSOT10）和串行 I/O 移位寄存器 10 （SIO10）被重置。

16.4.2 3 线串行I/O模式

使用 3 线串行 I/O 模式通过时钟驱动的串行接口来连接外部 IC 和显示控制器。

该模式中使用三条线进行通信： 串行时钟（$\overline{\text{SCK10}}$）、串行输出（SO10）和串行输入（SI10）线。

（1） 使用的寄存器

- 串行操作模式寄存器 10（CSIM10）
- 串行时钟选择寄存器 10（CSIC10）
- 端口模式寄存器 1（PM1）
- 端口寄存器 1（P1）

3 线串行 I/O 模式设置操作的基本步骤如下。

<1>　设置 CSIC10 寄存器（参见 图 **16-3**）
<2>　设置 CSIM10 寄存器的 0 第 4 和 6 位（DIR10 和 TRMD10）（参见 图 **16-2**）。
<3>　CSIM10 的第 7 位（CSIE1n）设置为 1。　→ 使能发送/接收。
<4>　将数据写入发送缓冲寄存器 10（SOTB10）。　　→ 启动发送/接收。
　　从串行 I/O 移位寄存器 10（SIO10）中读取数据。　→ 启动数据接收。

注意事项：　在设置端口模式寄存器和端口寄存器时，要考虑与通信另一方的关系。

寄存器设置与引脚之间的关系如下所示。

表 16-2. 寄存器设置与引脚之间的关系

CSIE10	TRMD10	PM12	P12	PM13	P13	PM11	P11	CSI10 操作	引脚功能		
									SI10/RxD0/<RxD6>/P12	SO10/TxD0/<TxD6>/P13	$\overline{SCK10}$/P11
0	×	×[注1]	×[注1]	×[注1]	×[注1]	×[注1]	×[注1]	停止	RxD0/<RxD6>/P12	TxD0/<TxD6>/P13	P11[注2]
1	0	1	×	×[注1]	×[注1]	1	×	从设备接收[注3]	SI10	TxD0/<TxD6>/P13	$\overline{SCK10}$（输入）[注3]
1	1	×[注1]	×[注1]	0	0	1	×	从设备发送[注3]	RxD0/<RxD6>/P12	SO10	$\overline{SCK10}$（输入）[注3]
1	1	1	×	0	0	1	×	从设备发送/接收[注3]	SI10	SO10	$\overline{SCK10}$（输入）[注3]
1	0	1	×	×[注1]	×[注1]	0	1	主设备接收	SI10	TxD0/<TxD6>/P13	$\overline{SCK10}$（输出）
1	1	×[注1]	×[注1]	0	0	0	1	主设备发送	RxD0/<RxD6>/P12	SO10	$\overline{SCK10}$（输出）
1	1	1	×	0	0	0	1	主设备发送/接收	SI10	SO10	$\overline{SCK10}$（输出）

注　**1.**　可被设置作为端口功能。
　　2.　如果 P10/$\overline{SCK10}$/TxD0 被用作端口引脚，则将 CKP10 清除为 0。
　　3.　要使用从设备模式，则须将 CKS102、CKS101 和 CKS100 设置为 1，1，1。

备注　×：　无须理会
　　CSIE10：　串行操作模式寄存器 10 （CSIM10）的第 7 位
　　TRMD10：　CSIM10 的第 6 位
　　CKP10：　串行时钟选择寄存器 10 （CSIC10）的第 4 位
　　CKS102，CKS101，CKS100：CSIC10 的第 0 位至第 2 位
　　PM1×：　端口模式寄存器
　　P1×：　端口输出锁存器

（2） 通信操作

在 3 线串行 I/O 模式下，数据的发送或接收以 8 位为单元。数据的每一位随着串行时钟被同步发送或接收。

当串行操作模式寄存器 10（CSIM10）的第 6 位（TRMD10）为 1 时，可以发送或接收数据。当数据被写入到发送缓冲寄存器 10（SOTB10）时，启动发送/接收。此外，当串行操作模式寄存器 10 （CSIM10）的第 6 位（TRMD10）为 0 时，可以接收数据。

当数据从串行 I/O 移位寄存器 10（SIO10）被读出时，启动接收操作。

通信开始后，CSIM10 的第 0 位（CSOT1n）被设置为 1。当 8 位数据通信结束时，设置通信完成中断请求标志（CSIIF10），且清除 CSOT10。这样使能下一次通信。

注意事项　当 CSOT10 = 1 时（串行通信期间），不要访问控制寄存器和数据寄存器。

图 16-6. 3 线串行 I/O 模式中的时序（1/2）

（a）　发送/接收时序 （类型 1：　TRMD10 = 1，DIR10 = 0，CKP10 = 0，DAP10 = 0）

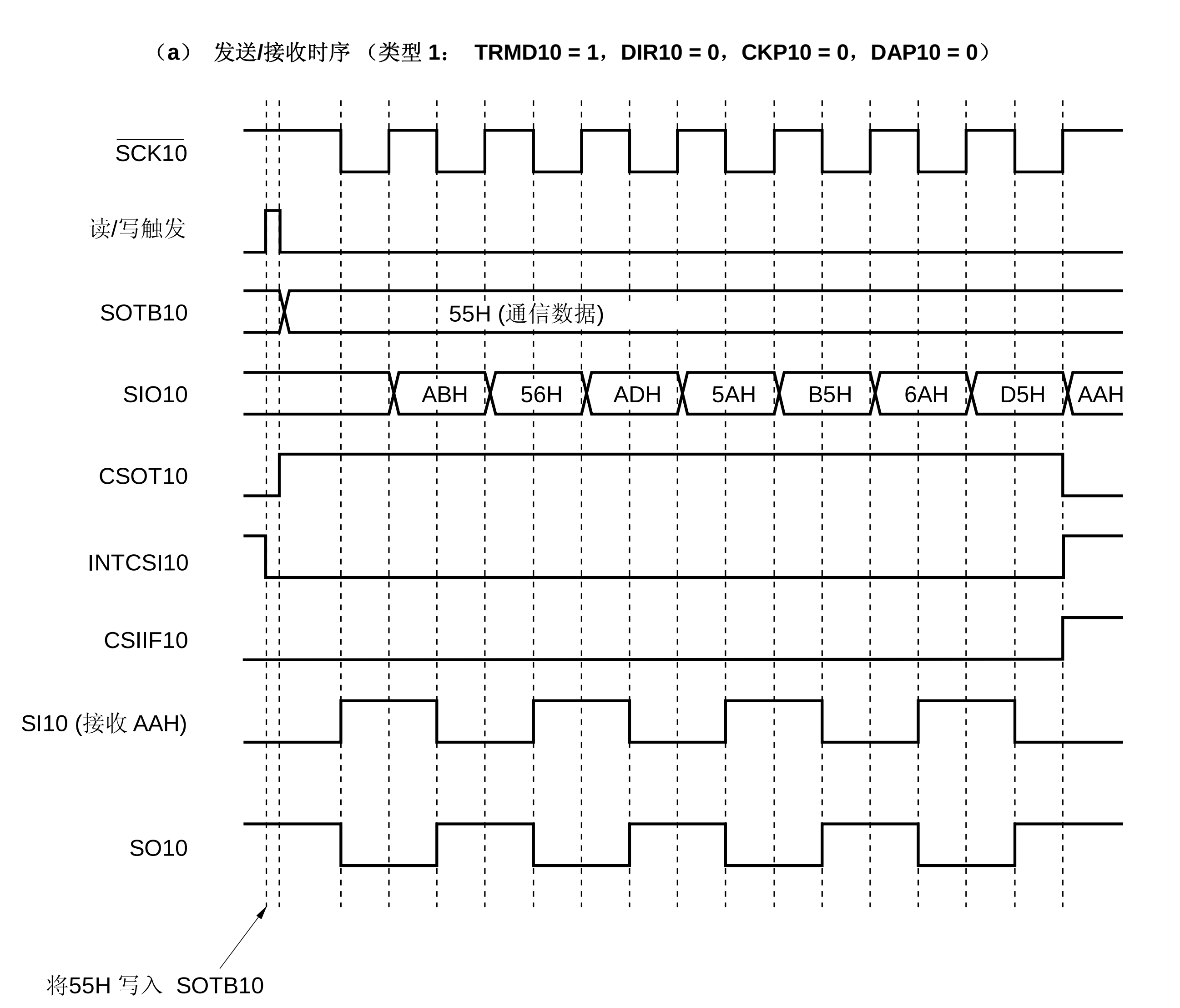


图 **16-6. 3** 线串行 **I/O** 模式中的时序（**2/2**）

（**b**） 发送/接收时序 （类型 **2**： **TRMD10 = 1**，**DIR10 = 0**，**CKP10 = 0**，**DAP10 = 1**）

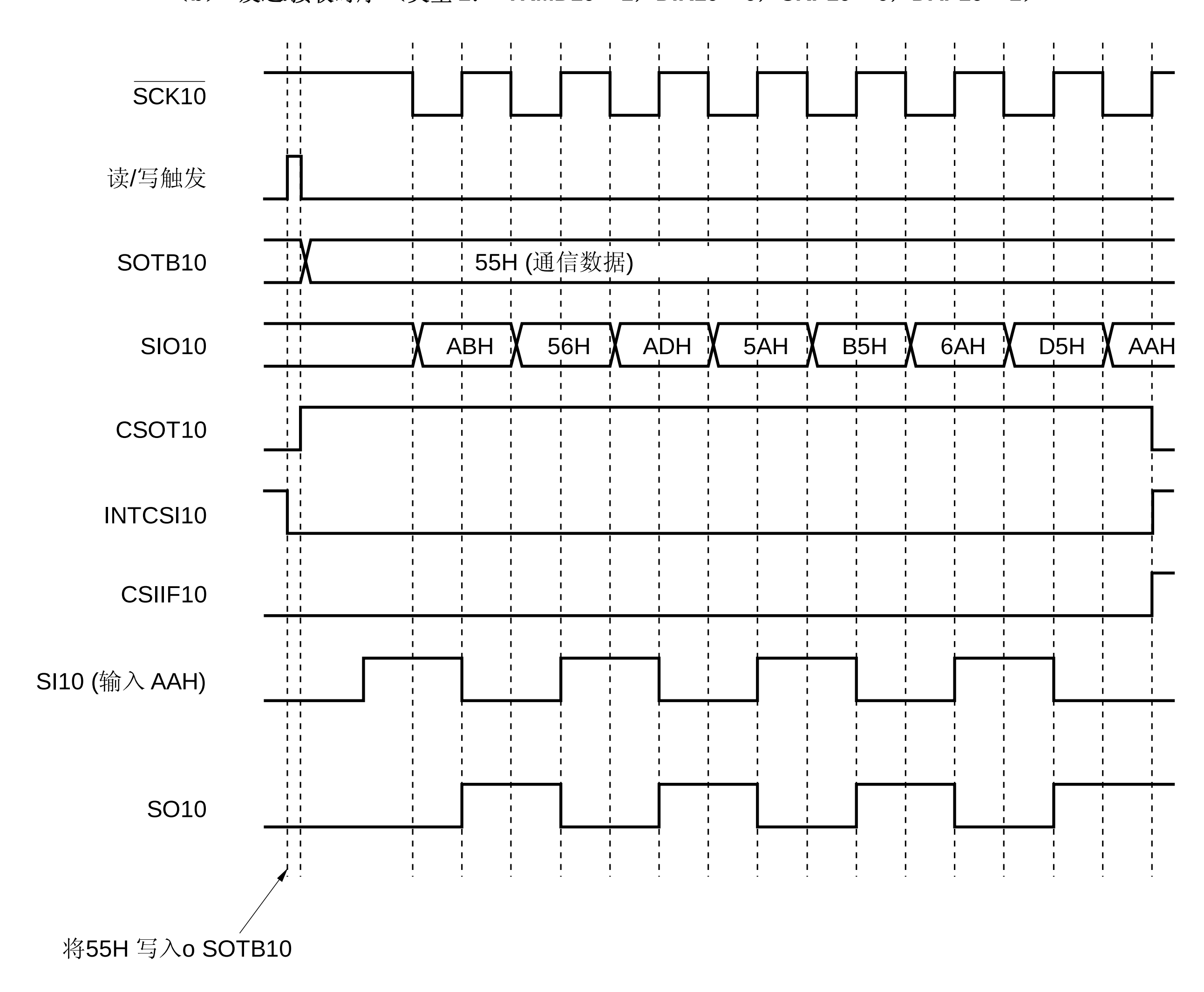
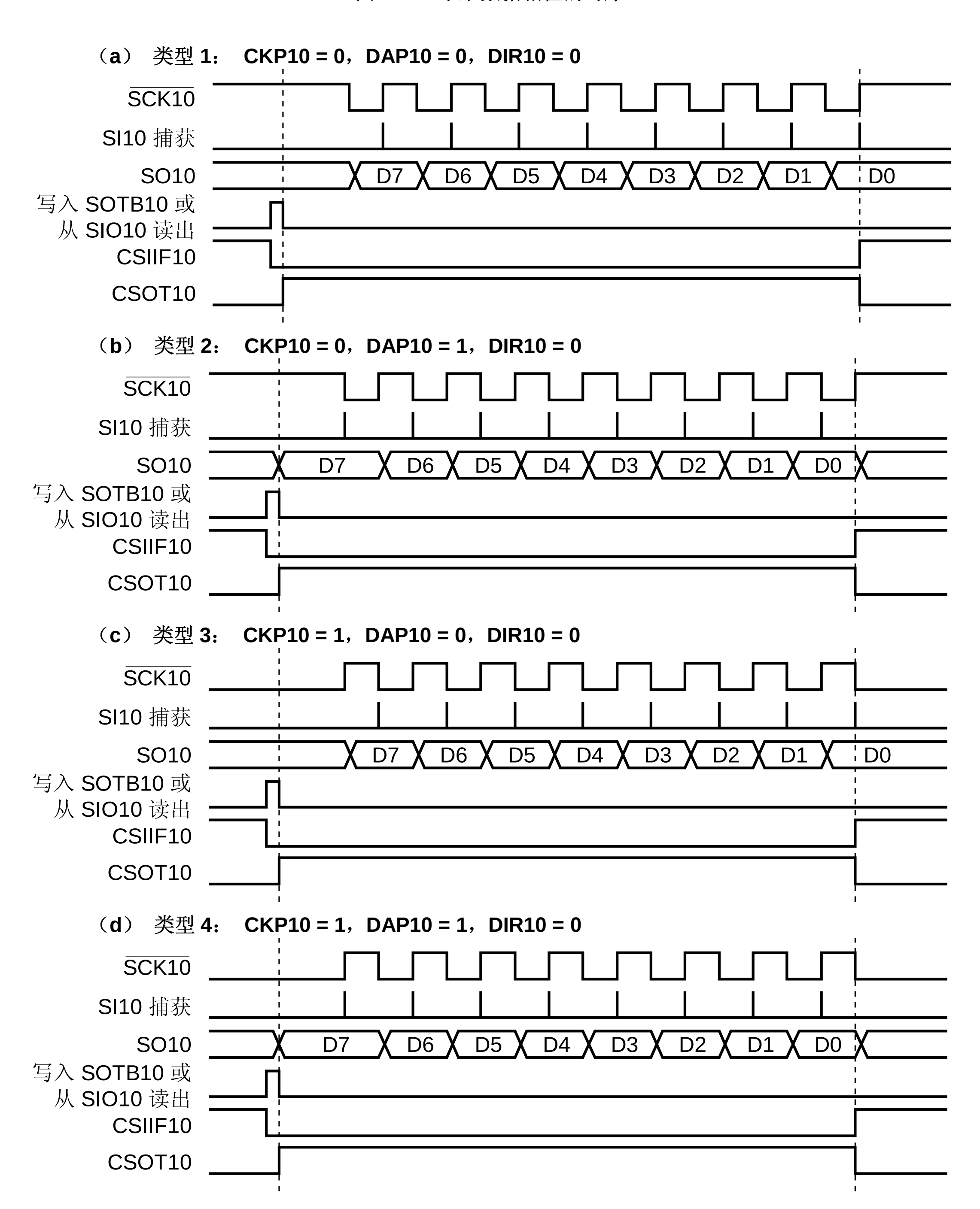


图 **16-7.** 时钟/数据相位的时序



备注　上图展示了一个通信操作过程，采用 MSB 先行方式发送数据。

（3） 输出到 SO10 引脚的时序（起始位）

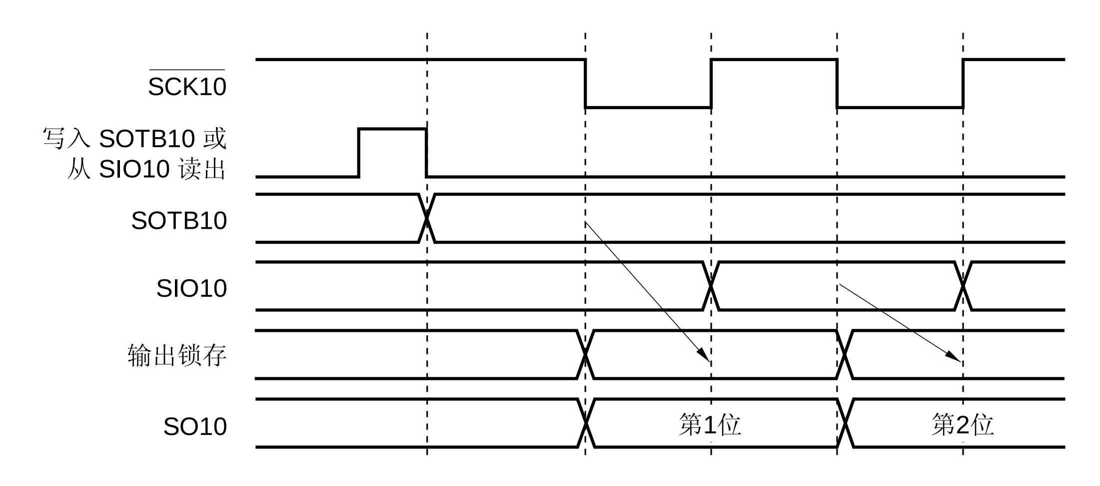
通信开始后，将发送缓冲寄存器 10（SOTB10）的值从 SO10 引脚输出。

此时起始位的输出操作描述如下。

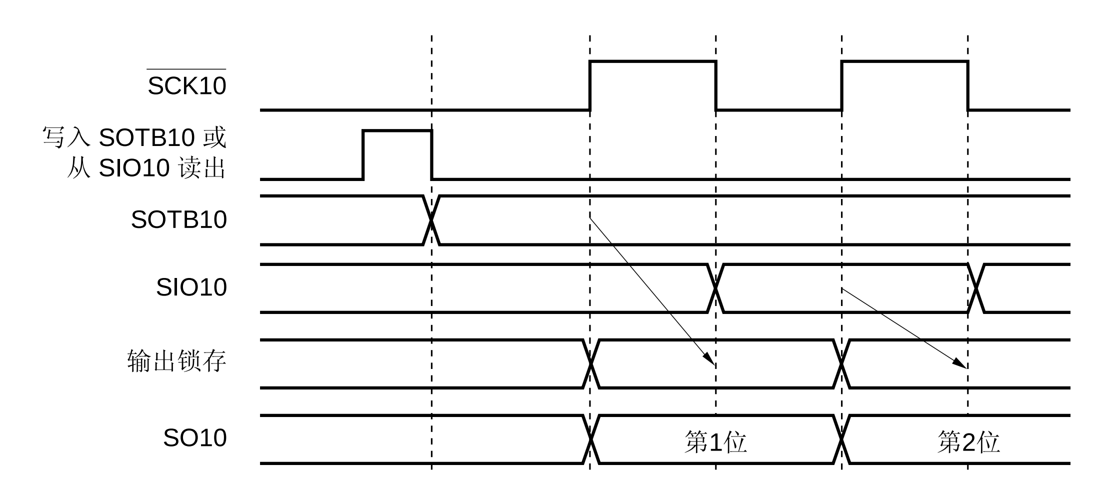
图 16-8. 起始位的输出操作（1/2）



（a） 类型 1： CKP10 = 0，DAP10 = 0



（b） 类型 3： CKP10 = 1，DAP10 = 0

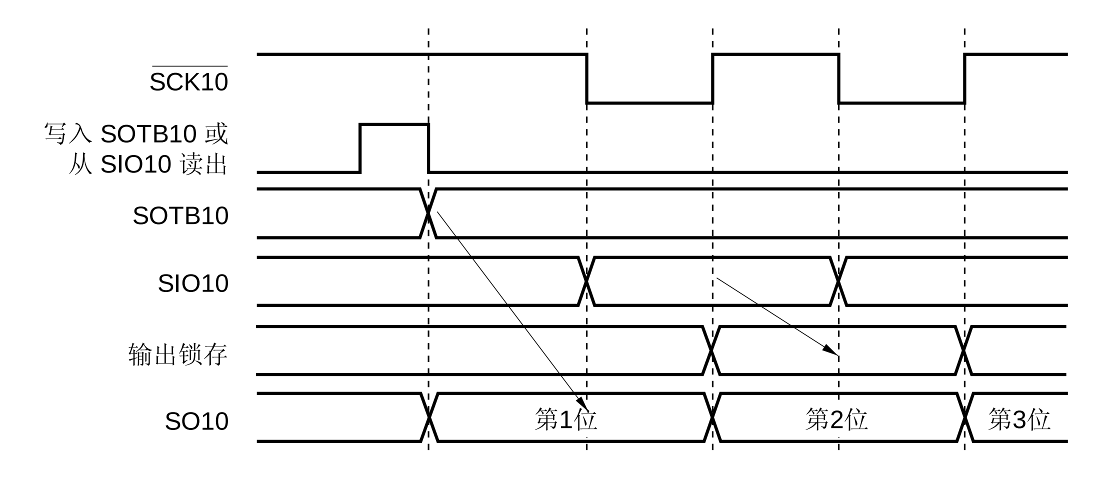


在 $\overline{\text{SCK10}}$ 的下降沿（或上升沿），起始位直接被 SOTB10 寄存器锁存到输出锁存器，并通过输出选择器从 SO10 引脚输出。然后在 $\overline{\text{SCK10}}$ 下一个上升沿（或下降沿）时将 SOTB10 寄存器的内容传送到 SIO10 寄存器，并移位 1 次。同时，将 SI10 引脚接收数据的起始位存储到 SIO10 寄存器。

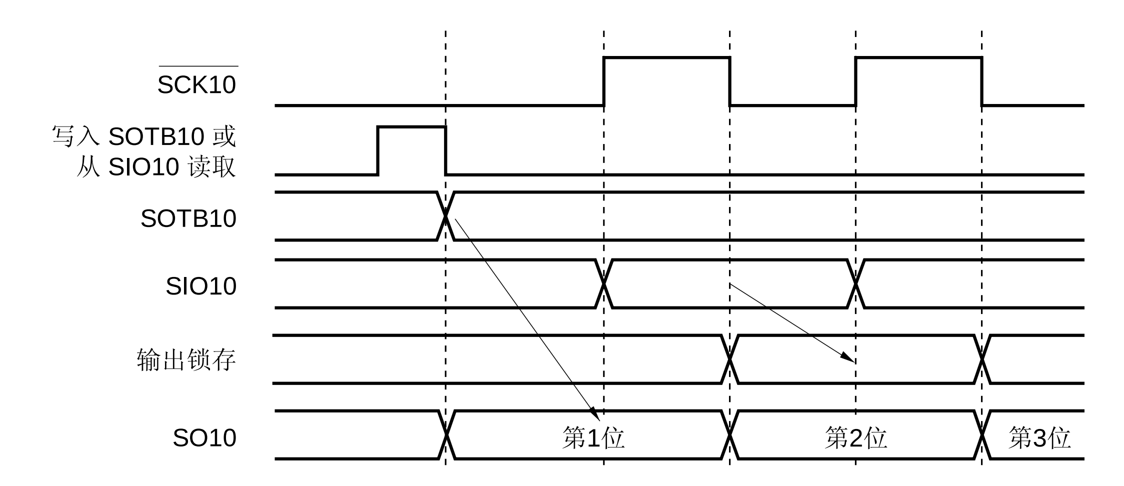
在下一个 $\overline{\text{SCK10}}$ 的下降沿（或上升沿），通过 SIO10 寄存器将第 2 位以及随后各位锁存到输出锁存器，然后数据从 SO10 引脚输出。

图 16-8. 起始位的输出操作（2/2）

（c） 类型 2： CKP10 = 0，DAP10 = 1



（d） 类型 4： CKP10 = 1，DAP10 = 1



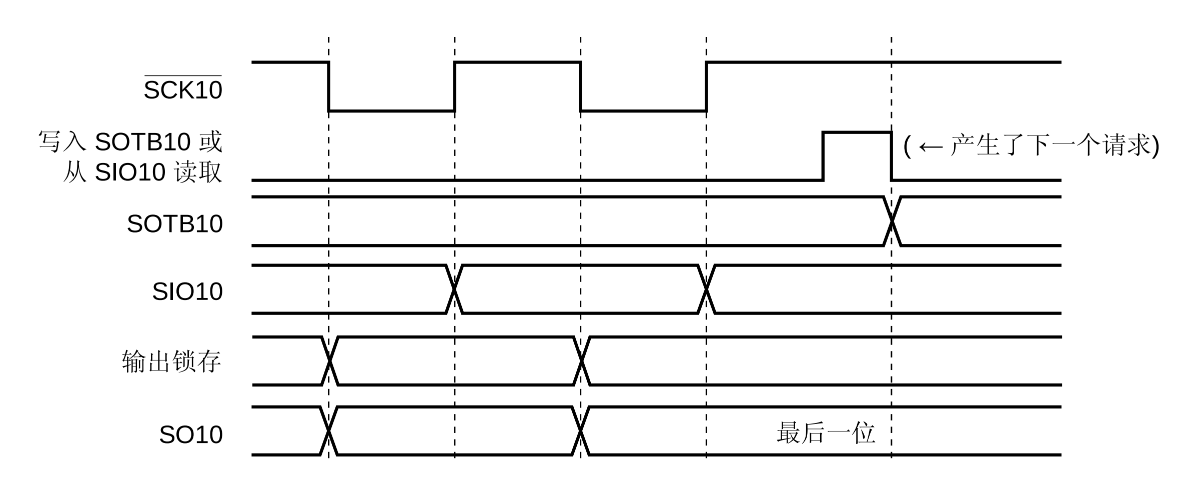
在 SOTB10 寄存器的写入信号或 SIO10 寄存器的读取信号的下降沿，起始位数据直接被 SOTB10 寄存器锁存，并通过输出选择器从 SO10 引脚输出。然后在 $\overline{SCK10}$ 下一个下降沿（或上升沿）时，将 SOTB10 寄存器的内容传送到 SIO10 中，并移位 1 次。同时，将 SI10 引脚接收数据的起始位存储到 SIO10 寄存器。
在下一个 $\overline{SCK10}$ 的上升沿（或下降沿），通过 SIO10 寄存器将第 2 位以及随后各位锁存到输出锁存器，然后数据从 SO10 引脚输出。

（4）**SO10** 引脚的输出值（最后一位）

在通信已经完成后，**SO10** 引脚保存最后一位的输出值。

图 **16-9. SO10** 引脚的输出值（最后一位）（**1/2**）

（**a**）类型 **1**：　**CKP10 = 0**，**DAP10 = 0**



（**b**）类型 **3**：　**CKP10 = 1**，**DAP10 = 0**

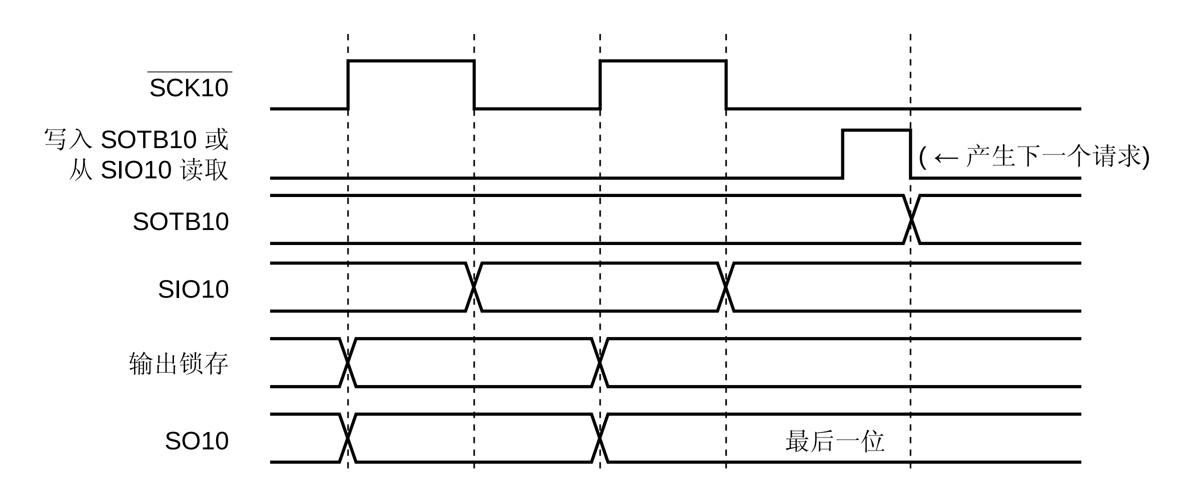
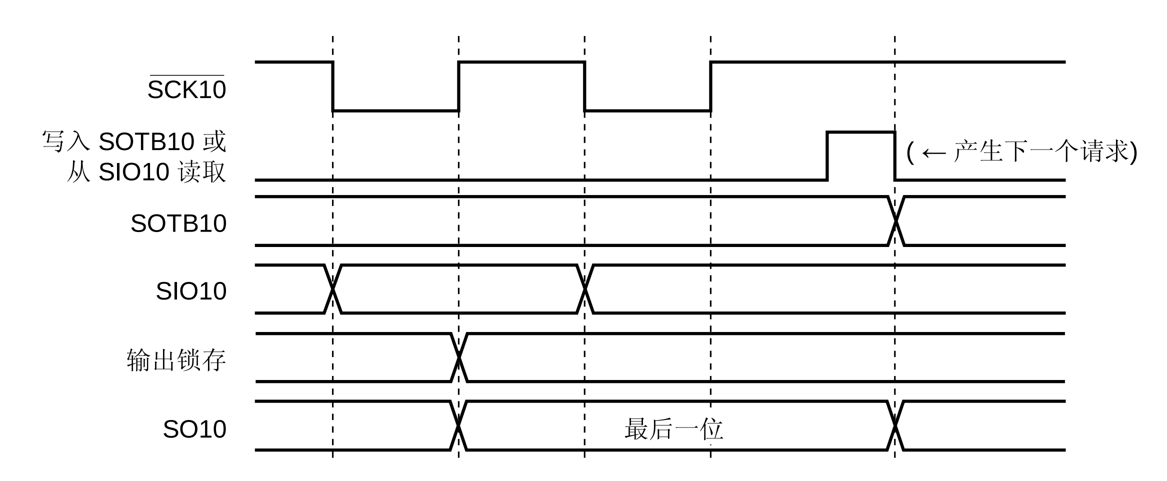
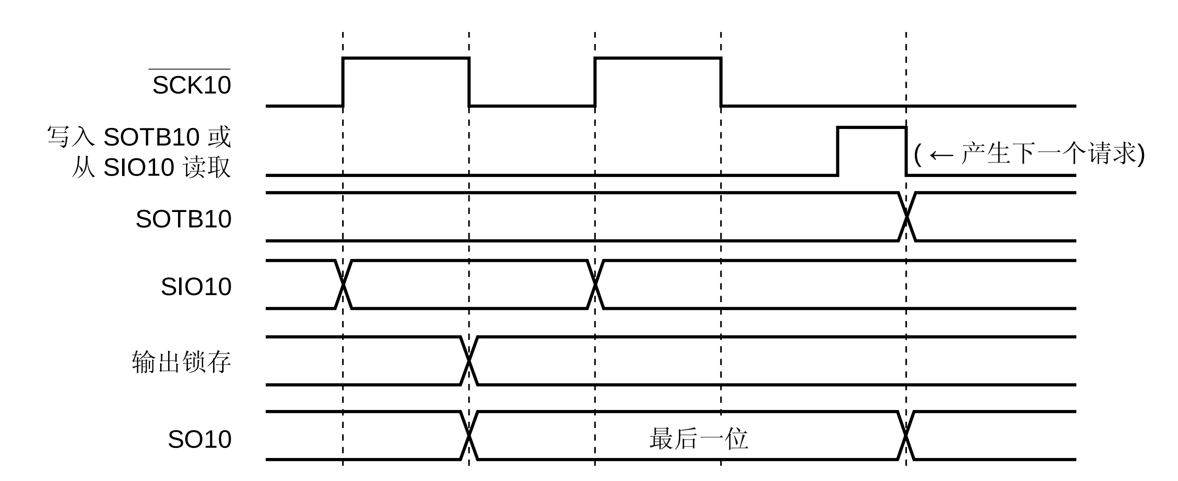


图 16-9. SO10 引脚的输出值（最后一位）（2/2）

（c） 类型 2： CKP10 = 0，DAP10 = 1



（d） 类型 4： CKP10 = 1，DAP10 = 1



（5） SO10 输出 （参考图 16-1）

如果串行操作模式寄存器 10（CSIM10）的第 7 位（CSIE1n）被清除为 0，则 SO10 输出的状态如下。

表 16-3. SO10 输出状态

TRMD10	DAP10	DIR10	SO10 输出注 1
TRMD10 = 0 注 2	–	–	输出低电平注 2
TRMD10 = 1	DAP10 = 0	–	SO10 锁存值 （低电平输出）
	DAP10 = 1	DIR10 = 0	SOTB10 第 7 位的值
		DIR10 = 1	SOTB10 第 0 位的值

注　**1.** 根据 PM12 和 P12 以及 SO10 的输出，决定 SO10/P12 引脚的实际输出。

2. 复位之后的状态。

注意事项　**如果有值被写入 TRMD10、DAP10 和 DIR10，则 SO10 的输出值将发生改变。**

17.1 LCD 控制器/驱动器的功能

78K0/LE3 的 LCD 控制器/驱动器具有如下功能。

（1） LCD 驱动器电压发生器可以切换外部电阻分压和内部电阻分压。
（2） 基于自动读取存储器显示数据，自动输出 segment 和公共端信号。
（3） 6 种不同的显示模式：
- 静态
- 1/2 占空比（1/2 偏压）
- 1/3 占空比（1/2 偏压）
- 1/3 占空比（1/3 偏压）
- 1/4 占空比（1/3 偏压）
- 1/8 占空比（1/4 偏压）

（4） 每种显示模式都有 6 种不同的帧频率。
（5） μPD78F044x： Segment 信号输出： 32 注 （SEG0 至 SEG31），
公共端信号输出： 8 注 （COM0 至 COM7）
μPD78F045x： Segment 信号输出： 32 注 （SEG0 至 SEG31），
公共端信号输出： 8 注 （COM0 至 COM7）
μPD78F046x： Segment 信号输出： 24 注 （SEG0 至 SEG23），
公共端信号输出： 8 注 （COM0 至 COM7）
（6） 在每种显示模式下（除静态模式外），LCD segment 信号的输出和 segment 键源信号的时间分频输出。
Segment 键源信号输出：最大值 8（SEG16（KS0） 至 SEG23（KS7））

注 4 个 segment 信号输出（SEG0 至 SEG3）和 4 个公共端信号输出（COM4 至 COM7）是复用功能引脚。仅当由 LCD 显示模式寄存器（LCDM）设置为 8 分时模式时，可以使用 COM4 至 COM7 。

表 17-1 列出了每种显示模式的最大显示像素数目。

表 17-1. 最大像素数目列表

（a） μPD78F044x，78F045x

LCD 驱动器电压发生器	偏压模式	时间片的数量	使用的公共端信号	Segment 数量	最大像素的数量
• 外部电阻分压 • 内部电阻分压	–	静态	COM0 （ COM1 至 COM3）	32	32（32 segment 信号，1 公共端信号）[注 2]
	1/2	2 [注 1]	COM0，COM1		64（32 segment 信号，2 公共端信号）[注 3]
		3 [注 1]	COM0 至 COM2		96（32 segment 信号，3 公共端信号）[注 4]
	1/3	3 [注 1]	COM0 至 COM2		
		4 [注 1]	COM0 至 COM3		128（32 segment 信号，4 公共端信号）[注 5]
	1/4	8 [注 1]	COM0 至 COM7	28	224（28 segment 信号，8 公共端信号）[注 6]

注　**1.** 使用 segment 键扫描功能（KSON = 1）时，为 segment 键扫描信号输出增加“时间片的数量 + 1”。
2. 4 个数字 LCD 面版，每个数字有 8-segment [illegible]配置。
3. 8 个数字 LCD 面版，每个数字有 4-segment [illegible]配置。
4. 12 个数字 LCD 面版，每个数字有 3-segment [illegible]配置。
5. 16 个数字 LCD 面版，每个数字有 2-segment [illegible]配置。
6. 28 个数字 LCD 面版，每个数字有 1-segment [illegible]配置。

（b） μPD78F046x

LCD 驱动器电压发生器	偏压模式	时间片的数量	使用的公共端信号	Segment 数量	最大像素的数量
• 外部电阻分压 • 内部电阻分压	–	静态	COM0 （ COM1 至 COM3）	24	24（24 segment 信号，1 公共端信号）[注 2]
	1/2	2 [注 1]	COM0，COM1		48（24segment 信号，2 公共端信号）[注 3]
		3 [注 1]	COM0 至 COM2		72（24 segment 信号，3 公共端信号）[注 4]
	1/3	3 [注 1]	COM0 至 COM2		
		4 [注 1]	COM0 至 COM3		96（24 segment 信号，4 公共端信号）[注 5]
	1/4	8 [注 1]	COM0 至 COM7	20	160（20 segment 信号，8 公共端信号）[注 6]

注　**1.** 使用 segment 键扫描功能（KSON = 1）时，为 segment 键扫描信号输出增加“时间片的数量 + 1”。
2. 3 个数字 LCD 面板，每个数字有 8-segment [illegible]配置。
3. 4 个数字 LCD 面板，每个数字有 4-segment [illegible]配置。
4. 9 个数字 LCD 面板，每个数字有 3-segment [illegible]配置。
5. 12 个数字 LCD 面板，每个数字有 2-segment [illegible]配置。
5. 20 个数字 LCD 面板，每个数字有 1-segment [illegible]配置。

17.2 LCD 控制器/驱动器 的配置

LCD 控制器/驱动器由以下硬件组成。

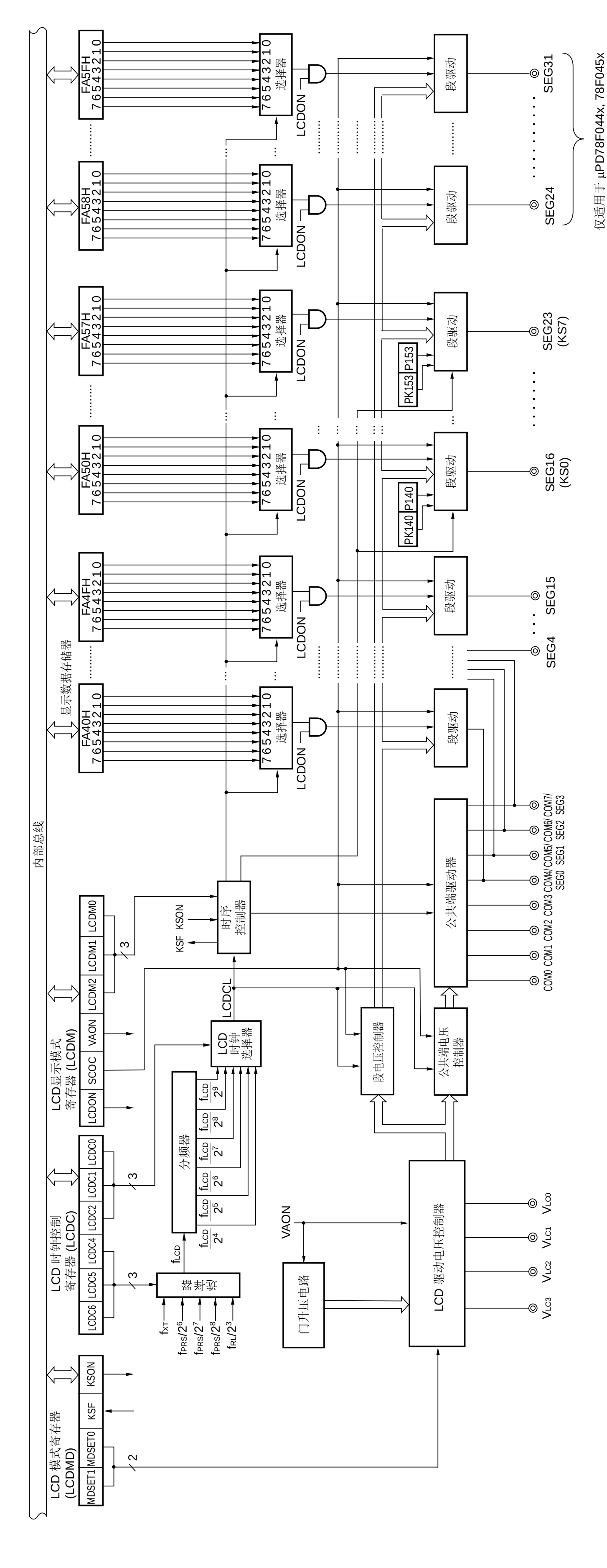
表 17-2. LCD 控制器/驱动器的配置

项目	配置
显示输出	μPD78F044x： 32 segment 信号[注 1]（SEG0-SEG31），8 公共端信号[注 1]（COM0 至 COM7） μPD78F045x： 32 segment 信号[注 1]（SEG0-SEG31），8 公共端信号[注 1]（COM0 至 COM7） μPD78F046x： 24 segment 信号[注 1]（SEG0-SEG23），8 公共端信号[注 1]（COM0 至 COM7）
Segment 键源输出	Segment 键源信号：8（SEG16（KS0）-SEG23（KS7））
控制寄存器	LCD 模式寄存器 （LCDMD） LCD 显示模式寄存器 （LCDM） LCD 时钟控制寄存器 （LCDC0） 端口功能寄存器 2 （PF2）[注 2] 端口功能寄存器 ALL （PFALL） 按键返回模式寄存器（KRM） 端口模式寄存器 4（PM4） 上拉电阻选项寄存器 4（PU4） 端口寄存器 14（P14） 端口寄存器 15（P15）

注　**1.** 4 个 segment 信号输出（SEG0 至 SEG3）和 4 个公共端信号输出（COM4 至 COM7）是复用功能引脚。仅当由 LCD 显示模式寄存器（LCDM）设置为 8 分时模式时，可以使用 COM4 至 COM7 。

2. 仅限μPD78F044x 和 78F045x。

图 17-1. LCD 控制器/驱动器的框图



17.3 寄存器控制LCD 控制器/驱动器

下面十个寄存器用于控制 LCD 控制器/驱动器。

- LCD 模式寄存器 （LCDMD）
- LCD 显示模式寄存器 （LCDM）
- LCD 时钟控制寄存器（LCDC0）
- 端口功能寄存器 2 （PF2）[注]
- 端口功能寄存器 ALL（PFALL）
- 按键返回模式寄存器（KRM）
- 端口模式寄存器 4（PM4）
- 上拉电阻选项寄存器 4（PU4）
- 端口寄存器 14（P14）
- 端口寄存器 15（P15）

注　仅限μPD78F044x 和 78F045x。

（1） LCD 模式寄存器 （LCDMD）

LCDMD 用于设置 LCD 驱动电压发生器。

可以使用 1 位或 8 位存储器操作指令来设置 LCDMD。

复位信号的发生会将 LCDMD 设置为 00H。

图 17-2. LCD 显示模式寄存器的格式

地址： FFB0H　复位后： 00H　R/W[注 1]

符号	7	6	5	4	3	2	1	0
LCDMD	0	0	MDSET1	MDSET0	0	0	KSF	KSON

MDSET1	MDSET0	LCD 驱动电压发生器的选择
0	0	外部电阻分压方式，内部电阻断开连接
0	1	内部电阻分压方式，内部电阻连接 （无减压转换，当 $V_{LCD} = V_{DD}$ 时使用）
1	1	内部电阻分压方式，内部电阻连接 （减压转换，当 $V_{LCD} = 3/5V_{DD}$ 时使用）
其他情况		禁止设置

KSF	Segment 键扫描状态
0	LCD 显示信号正在输出
1	Segment 键扫描信号正在输出

KSON	Segment 键扫描功能控制
0	不使用Segment 键扫描功能
1	使用Segment 键扫描 功能[注 2]

注　**1.** 第 1 位是只读的。

　　2. 如果 V_{DD} 等于 V_{LC0}，使用 segment 键扫描功能。

　　只有 KR0 至 KR4 引脚可以被用作 segment 键扫描功能的输入引脚。

注意事项　第 0 位至第 2 位，第 3 位，第 6 位和第 7 位必须被设置为 0。

（2） LCD 显示模式寄存器（LCDM）

LCDM 指定是否使能显示操作。它也可以指定是否使能 segment 引脚/公共端引脚输出，门升压电路控制，和显示模式。

可以使用 1 位或 8 位存储器操作指令来设置 LCDM。

复位信号的发生会将 LCDM 设置为 00H。

图 17-3. LCD 显示模式寄存器的格式

地址： FFB1H 地址： 00H R/W

符号	<7>	<6>	5	<4>	3	2	1	0
LCDM	LCDON	SCOC	0	VAON	0	LCDM2	LCDM1	LCDM0

LCDON	LCD显示使能/禁止
0	显示关闭（全部segment 输出都未选）
1	显示打开

SCOC	Segment引脚/公共端引脚输出控制[注 1]
0	输出低电平到segment/公共端引脚
1	输出未选电平到segment 引脚，输出LCD波形到公共端引脚

VAON	门升压电路控制[注1，2]
0	门电压不升压
1	门电压升压

LCDM2	LCDM1	LCDM0	LCD 控制器/驱动器显示模式选择	
			电阻分压方式	
			时间片的数量	偏压模式
1	1	1	8 [注 3]	1/4 [注 4]
0	0	0	4 [注 3]	1/3
0	0	1	3 [注 3]	1/3
0	1	0	2 [注 3]	1/2
0	1	1	3 [注 3]	1/2
1	0	0	静态	
其他值			禁止设置	

（注和注意事项在下页列出。）

注 **1.** 不执行或不需要 LCD 显示时，使用下列设置，可以降低功耗。
<1> 将 SCOC 和 VAON 都设置为 0，为了降低功耗。
<2> 使用内部电阻分压方式时，认为 MDSET1，MDSET0 =（0，0）.
（可以减少流经内部电阻的电流。）

2. 该位用于控制 LCD 控制器/驱动器的内部门信号的升压。
如果设置为“内部门电压升压”，LCD 驱动性能增强。
根据如下条件设置 VAON。
<设置为静态显示模式时>
- 当 2.0 V ≤ V_{LCD} ≤ V_{DD} ≤ 5.5 V： VAON = 0
- 当 1.8 V（ VLCD（ VDD（ 3.6 V： VAON = 1

<设置为 1/3 偏压方式时>
- 当 2.5 V（ VLCD（ VDD（ 5.5 V： VAON = 0
- 当 1.8 V（ VLCD（ VDD（ 3.6 V： VAON = 1

<设置为 1/2 偏压方式或 1/4 偏压方式时>
- 当 2.7 V ≤ V_{LCD} ≤ V_{DD} ≤ 5.5 V： VAON = 0
- 当 1.8 V ≤ V_{LCD} ≤ V_{DD} ≤ 3.6 V： VAON = 1

3. 使用 segment 键扫描功能（KSON = 1）时，为 segment 键扫描信号输出增加“时间片的数量 + 1”。

4. 当 P40/KR0/V_{LC3} 引脚被设置为 1/4 偏压方式时，它被用作 V_{LC3}。当该引脚被设置为另外一种偏压方式时，它被用作端口功能（P40） 或按键中断功能（KR0）。

注意事项 **1. 第 3 位和第 5 位必须被设置为 0。**

2. 当在一个使用较多 COM 的模式下显示时，比如 8COM，在低电压条件下 V_{LC0} 可能无法获得足够的对比，取决于面板的特性。在已经执行全面 LCD 显示评估，并确认显示质量方面没有问题之后，使用这个 LCD 控制器/驱动器。

（3） LCD 时钟控制寄存器 （LCDC0）

LCDC0 用于指定 LCD 源时钟和 LCD 时钟。

根据 LCD 时钟和时间片的数量来决定帧频率。

可以使用 1 位或 8 位存储器操作指令来设置 LCDC0。

复位信号的发生会将 LCDC0 设置为 00H。

图 17-4. LCD 时钟控制寄存器的格式

地址： FFB2H 地址： 00H R/W

符号	7	6	5	4	3	2	1	0
LCDC0	0	LCDC6	LCDC5	LCDC4	0	LCDC2	LCDC1	LCDC0

LCDC6	LCDC5	LCDC4	LCD 源时钟（f_{LCD}）选择
0	0	0	f_{XT} （32.768 kHz）
0	0	1	$f_{PRS}/2^6$
0	1	0	$f_{PRS}/2^7$
0	1	1	$f_{PRS}/2^8$
1	0	0	$f_{RL}/2^3$
其他情况			禁止设置

LCDC2	LCDC1	LCDC0	LCD时钟（LCDCL）选择
0	0	0	$f_{LCD}/2^4$
0	0	1	$f_{LCD}/2^5$
0	1	0	$f_{LCD}/2^6$
0	1	1	$f_{LCD}/2^7$
1	0	0	$f_{LCD}/2^8$
1	0	1	$f_{LCD}/2^9$
其他情况			禁止设置

注意事项 第 3 位和第 7 位必须设置为 0。

备注 **1.** f_{XT}： XT1 时钟振荡频率

2. f_{PRS}： 外设硬件时钟频率

3. f_{RL}： 内部低速振荡时钟频率

（4） 端口功能寄存器 2 （PF2）（仅限(PD78F044x 和 78F045x）

该寄存器用于设置引脚 P20 至 P27 用作端口引脚（segment 输出引脚除外）或 segment 输出引脚。
可以使用 1 位或 8 位存储器操作指令来设置 PF2。
复位信号的发生会将 PF2 设置为 00H。

图 17-5. 端口功能寄存器 2 的格式

地址：FFB5H 地址：00H R/W

符号	7	6	5	4	3	2	1	0
PF2	PF27	PF26	PF25	PF24	PF23	PF22	PF21	PF20

PF2n	端口/segment 输出规范
0	用作端口（segment输出除外）
1	用于segment输出

备注 n = 0 至 7

（5） 端口功能寄存器 ALL（PFALL）

该寄存器设置引脚 P8，P10，P11，P14 和 P15 用作端口引脚 （segment 输出除外）或 segment 输出引脚。
可以使用 1 位或 8 位存储器操作指令来设置 PFALL。
复位信号的发生会将 PFALL 设置为 00H。

图 17-6. 端口功能寄存器 ALL 的格式

地址：FFB6H 地址：00H R/W

符号	7	6	5	4	3	2	1	0
PFALL	0	PF15ALL	PF14ALL	0	PF11ALL	PF10ALL	0	PF08ALL

PFnALL	端口/segment 输出选择
0	用作端口（segment输出除外）
1	用于segment输出

备注 n = 08，10，11，14，15

（6） 按键返回模式寄存器（KRM）

当使用 segment 键扫描功能时，该寄存器用于指定一个引脚用作 segment 键扫描输入引脚。

不使用 segment 键扫描功能时，参见图 21-2 按键返回模式寄存器（KRM）的格式。

可以使用 1 位或 8 位存储器操作指令来设置 KRM。

复位信号的发生会将该寄存器设置为 00H。

图 17-7. 按键返回模式寄存器（KRM）的格式

地址： FF6EH 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
KRM	0	0	0	KRM4	KRM3	KRM2	KRM1	KRM0

KRMn	segment 键扫描输入引脚的设置（n = 0至4）
0	不将指定引脚用作segment 键扫描输入引脚.
1	指定引脚用作segment 键扫描输入引脚

注意事项 **1.** 如果 **KRM** 被改变，中断请求信号可能会被置位。因此，禁止中断，然后改变 **KRM** 寄存器。清除中断请求信号并使能中断。

2. 设置不将指定引脚用作 **segment** 键扫描输入引脚（**KRMn = 0**），对应的 **P4n** 引脚可以用作普通端口。

3. 当使用 **P40/KR0/V**$_{LC3}$ 引脚作为按键中断功能（**KR0**），设置 **LCD** 显示模式寄存器（**LCDM**）选择 **1/4** 偏压方式以外的设置。当设置为 **1/4** 偏压方式时，**P40/KR0/V**$_{LC3}$ 引脚作为 **V**$_{LC3}$。

（7）　端口模式寄存器 4（PM4）

该寄存器用于按位设置端口 4 的输入/输出模式。

当使用 segment 键扫描功能时，为了设置 P4n 引脚作为键扫描输入引脚，被使用端口的端口模式寄存器设置为 1（PM4n = 1）。

可以通过 1 位或 8 位存储器操作指令来设置 PM4。

复位信号的产生会将该寄存器设置为 FFH。

图 17-8. 端口模式寄存器 4（PM4）的格式

地址：FF24H　复位后：FFH　R/W

符号	7	6	5	4	3	2	1	0
PM4	1	1	1	PM44	PM43	PM42	PM41	PM40

PF4n	P4n 引脚I/O 模式的选择（n = 0至4）
0	输出模式（输出缓冲打开）
1	输入模式（输出缓冲关闭）

（8）　上拉电阻选项寄存器 4（PU4）

该寄存器用于是否使用 P40 至 P44 的片上上拉电阻。

当使用 segment 键扫描功能时，为了设置 P4n 引脚作为键扫描输入引脚，被使用端口的上拉电阻选项寄存器设置为 0（PU4n = 0）。

可以通过 1 位或 8 位存储器操作指令来设置 PU4。

复位信号的产生会将该寄存器设置为 00H。

图 17-9. 上拉电阻选项寄存器 4 的格式

地址：FF34H　复位后：00H　R/W

符号	7	6	5	4	3	2	1	0
PU4	0	0	0	PU44	PU43	PU42	PU41	PU40

PF4n	P4n 引脚片上上拉电阻的选择（n = 0至4）	
	引脚使用 segment 键扫描功能	引脚未使用segment 键扫描功能
0	只有在segment 键扫描输出期间，连接片上上拉电阻	不连接片上上拉电阻
1	禁止设置	连接片上上拉电阻

（9）　端口寄存器 14（P14）

当使用 segment 键扫描功能时，该寄存器使用第 0 位至第 3 位执行 KS0 至 KS3 输出前半部的控制，使用第 4 位至第 7 位执行 KS0 至 KS3 输出后半部的控制。

当 P14n 引脚用作 segment 键扫描输出时，P14n 和 PK14n 为都用来控制。

不使用 segment 键扫描功能时，参见 **图 4-22 端口寄存器的格式**

可以通过 1 位或 8 位存储器操作指令来设置 P14。

复位信号的产生会将该寄存器设置为 00H。

图 17-10. 端口寄存器 14（P14）的格式

地址：FF0EH　复位后：00H　R/W

符号	7	6	5	4	3	2	1	0
P14	PK143	PK142	PK141	PK140	P143	P142	P141	P140

P140-P143	KS0 至 KS3 输出前半部的控制
0	低电平输出
1	高电平输出

PK140-PK143	KS0 至 KS3 输出后半部的控制
0	低电平输出
1	高电平输出

（10）端口寄存器 15（P15）

当使用 segment 键扫描功能时，该寄存器使用第 0 位至第 3 位执行 KS4 至 KS7 输出前半部的控制，使用第 4 位至第 7 位执行 KS4 至 KS7 输出后半部的控制。

当 P15n 引脚用作 segment 键扫描输出时，P15n 和 PK15n 为都用来控制。

不使用 segment 键扫描功能时，参见 **图 4-22 端口寄存器的格式**

可以通过 1 位或 8 位存储器操作指令来设置 P15。

复位信号的产生会将该寄存器设置为 00H。

图 17-11. 端口寄存器 15（P15）的格式

地址：FF0FH　复位后：00H　R/W

符号	7	6	5	4	3	2	1	0
P15	PK153	PK152	PK151	PK150	P153	P152	P151	P150

P150-P153	KS4至KS7输出前半部的控制
0	低电平输出
1	高电平输出

PK150-PK153	KS4 至 KS7 输出后半部的控制
0	低电平输出
1	高电平输出

17.4 设置LCD 控制器/驱动器

17.4.1 不使用segment 键扫描功能（KSON = 0）时的设置方式

不使用 segment 键扫描功能（KSON = 0）时，LCD 控制器/驱动器的设置如下。通过下列步骤设置 LCD 控制器/驱动器。

<1> 设置（VAON = 1）内部门电压升压（LCD 显示模式寄存器（LCDM）的第 4 位）。[注]
<2> 通过 MDSET0 和 MDSET1 （LCD 模式寄存器（LCDMD）的第 4 和 5 位）设置电阻分压方式（MDSET0 = 0：外部电阻分压方式， MDSET0 = 1：内部电阻分压方式）。
<3> 设置用作 segment 输出的引脚到端口功能寄存器（PF2m，PFnALL）。
<4> 为了 LCD 显示，设置初始值到 RAM。
<5> 通过 LCDM0 至 LCDM2 （LCD 显示模式寄存器（LCDM）的第 0 位至第 2 位）设置时间片的数量。
<6> 通过 LCD 时钟控制寄存器 0（LCDC0）设置 LCD 源时钟和 LCD 时钟。
<7> 设置（SCOC = 1） SCOC （LCD 显示模式寄存器（LCDM）的第 6 位）。
未选信号从所有 segment 和公共端引脚输出，进入无显示状态。
<8> 通过设置 （LCDON = 1） LCDON （LCDM 的第 7 位），开始每个数据存储器对应的输出。

在该过程之后，设置待显示的数据到数据存储器。

注　根据如下条件设置 VAON。

<设置静态显示模式时>
- 当 2.0 V ≤ V_{LCD} ≤ V_{DD} ≤ 5.5 V： VAON = 0
- 当 1.8 V ≤ V_{LCD} ≤ V_{DD} ≤ 3.6 V： VAON = 1

<设置为 1/3 偏压方式时>
- 当 2.5 V ≤ V_{LCD} ≤ V_{DD} ≤ 5.5 V： VAON = 0
- 当 1.8 V ≤ V_{LCD} ≤ V_{DD} ≤ 3.6 V： VAON = 1

<设置为 1/2 偏压方式或 1/4 偏压方式时 >
- 当 2.7 V ≤ V_{LCD} ≤ V_{DD} ≤ 5.5 V： VAON = 0
- 当 1.8 V ≤ V_{LCD} ≤ V_{DD} ≤ 3.6 V： VAON = 1

备注　**1.** 使用下列过程来设置显示关闭状态，并断开使用内部电阻分压方式时的内部电阻
<1> 清除 LCDON（LCDM 的第 7 位）（LCDON = 0）.
未选信号从所有 segment 和公共端引脚输出，进入无显示状态。
<2> 清除 SCOC（LCD 显示模式寄存器（LCDM）的第 6 位）（SCOC = 0）
地电平从所有 segment 和公共端引脚输出。
<3> 假设 MDSET0，MDSET1（LCD 模式寄存器（LCDMD）的第 4 位和第 5 位） =（0，0），并设置电阻分压方式为外部电阻分压方式.

2. m = 0 至 7，n = 08，10，11，14，15

注意事项　**当在一个使用较多 COM 的模式下显示时，比如 8COM，在低电压条件下 VLC0 可能无法获得足够的对比，取决于面板的特性。在已经执行全面 LCD 显示评估，并确认显示质量方面没有问题之后，使用这个 LCD 控制器/驱动器。**

17.4.2 使用segment 键扫描功能（KSON = 1）时的设置方式

使用 segment 键扫描 功能（KSON = 1）时，LCD 控制器/驱动器的设置如下。通过下列步骤设置 LCD 控制器/驱动器。

<1> 设置（VAON = 1）内部门电压升压（LCD 显示模式寄存器（LCDM）的第 4 位）。[注 1]

<2> 通过 MDSET0 和 MDSET1 （LCD 模式寄存器（LCDMD）的第 4 和 5 位）设置电阻分压方式（MDSET0 = 0：外部电阻分压方式， MDSET0 = 1：内部电阻分压方式）。
设置（KSON = 1） KSON（LCD 模式寄存器（LCDMD）的第 0 位）

<3> 设置用作 segment 输出的引脚到端口功能寄存器（PF2m，PFnALL）。

<4> 被用作键扫描输入引脚[注 2] 的引脚，使用端口模式寄存器 4（PM4）设置为 PM4p = 1（输入模式）。

<5> 被用作键扫描输入引脚[注 2] 的引脚，使用上拉电阻选项寄存器 4（PU4） 设置为 PU4p = 0（只有在 segment 键扫描输出期间，连接片上上拉电阻）。

<6> 被用作 segment 键扫描输入引脚[注 2] 的引脚，使用按键返回模式寄存器（KRM） 设置为 KRMp = 1[注 3]。

<7> 为了 LCD 显示，设置初始值到 RAM。

<8> 设置 segment 键扫描输出的初始值到 P14，P15.

<9> 通过 LCDM0 至 LCDM2 （LCD 显示模式寄存器（LCDM）的第 0 位至第 2 位）设置时间片的数量。

<10> 通过 LCD 时钟控制寄存器 0（LCDC0）设置 LCD 源时钟和 LCD 时钟。

<11> 设置（SCOC = 1） SCOC （LCD 显示模式寄存器（LCDM）的第 6 位）。
未选信号从所有 segment 和公共端引脚输出，进入无显示状态。

<12> 通过设置 （LCDON = 1） LCDON （LCDM 的第 7 位），开始每个数据存储器对应的输出。

此后，根据要显示的内容，将数据设置到数据存储器，并根据 segment 键扫描输出的内容，执行端口寄存器（P14，P15）设置的 segment 键扫描输出。

注　**1.** 根据如下条件设置 VAON。

<设置为 1/3 偏压方式时>

- 当 $2.5\ V \le V_{LCD} \le V_{DD} \le 5.5\ V$： VAON = 0
- 当 $1.8\ V \le V_{LCD} \le V_{DD} \le 3.6\ V$： VAON = 1

<设置为 1/2 偏压方式或 1/4 偏压方式时 >

- 当 $2.7\ V \le V_{LCD} \le V_{DD} \le 5.5\ V$： VAON = 0
- 当 $1.8\ V \le V_{LCD} \le V_{DD} \le 3.6\ V$： VAON = 1

2. 使用 segment 键扫描功能时，请确保设置端口 4 作为 segment 键扫描输入引脚，且要使用的端口的上拉电阻选项寄存器设置为 PU4p = 0（只有在 segment 键扫描输出期间，连接片上上拉电阻）。
不能使用外部上拉电阻，因为它会影响 LCD 显示输出。

3. 如果 KRM 被改变，中断请求信号可能会被置位。因此，禁止中断，然后改变 KRM 寄存器。清除中断请求信号之后，使能中断。

备注　**1.** 使用下列过程来设置显示关闭状态，并断开使用内部电阻分压方式时的内部电阻

<1> 清除 LCDON（LCDM 的第 7 位）（LCDON = 0）.
未选信号从所有 segment 和公共端引脚输出，进入无显示状态。

<2> 清除 SCOC（LCD 显示模式寄存器（LCDM）的第 6 位）（SCOC = 0）
地电平从所有 segment 和公共端引脚输出。

<3> 假设 MDSET0，MDSET1（LCD 模式寄存器（LCDMD）的第 4 位和第 5 位）=（0，0），并设置电阻分压方式为外部电阻分压方式.

2. m = 0 至 7，n = 08，10，11，14，15，p = 0 至 4

注意事项　**当在一个使用较多 COM 的模式下显示时，比如 8COM，在低电压条件下 VLC0 可能无法获得足够的对比，取决于面板的特性。在已经执行全面 LCD 显示评估，并确认显示质量方面没有问题之后，使用这个 LCD 控制器/驱动器。**

17.5 LCD显示数据存储器

LCD 显示数据存储器是地址 FA40H 至 FA5FH（μPD78F044x 和 78F045x） 或 FA40H 至 FA57H（μPD78F046x）。使用 LCD 控制器/驱动器，可以将 LCD 显示数据存储器中的数据显示在 LCD 面板上。

图 17-12 显示了 LCD 显示数据存储器的内容和 segment/公共端输出之间的关系。

该区域不用于显示时，可用作普通 RAM。

图 **17-12. LCD** 显示数据存储器的内容和 **segment/**公共端输出之间的关系

（a） μPD78F044x，78F045x

	b7	b6	b5	b4	b3	b2	b1	b0	
FA67H									SEG39
FA66H									SEG38
FA65H									SEG37
⋮									
FA45H									SEG5
FA44H									SEG4
FA43H	0	0	0	0					SEG3
FA42H	0	0	0	0					SEG2
FA41H	0	0	0	0					SEG1
FA40H	0	0	0	0					SEG0
	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0	

（b） μPD78F046x

	b7	b6	b5	b4	b3	b2	b1	b0	
FA5FH									SEG31
FA5EH									SEG30
FA5DH									SEG29
⋮									
FA45H									SEG5
FA44H									SEG4
FA43H	0	0	0	0					SEG3
FA42H	0	0	0	0					SEG2
FA41H	0	0	0	0					SEG1
FA40H	0	0	0	0					SEG0
	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0	

注意事项 **FA40H** 至 **FA43H** 的高 **4** 位未分配存储器，请确保这些位为 **0**。

17.6 公共端和 Segment 信号

当对应的公共端和 segment 信号之间的电位差大于指定电压（LCD 驱动电压，V_{LCD}）时，LCD 面板上的每个像素被打开。当电位差低于 V_{LCD} 时，像素被关闭。

LCD 面板的公共端和 segment 信号使用 DC 电压，会导致性能衰退。为了避免这个问题，使用 AC 电压驱动 LCD 面板。

（1） 公共端信号

在表 17-3 列出的时序，根据指定的时间片数量，顺序选择各个公共端信号。在静态显示模式下，相同的信号输出到 COM0 至 COM3。

使用 segment 键扫描输出功能（KSON = 1）时，在一个 LCD 输出周期之后，segment 键扫描输出将会执行一个时间片的时期。此时产生的公共端信号在输出时不显示。

在 2 分时模式下，COM2 和 COM3 引脚开路悬空。在 3 分时模式中，COM3 引脚开路悬空。

在 8 分时模式之外，COM4 至 COM7 引脚悬空或用作 segment 引脚。

表 17-3. COM 信号

COM 信号 / 时间片的数量	COM0	COM1	COM2	COM3	COM4	COM5	COM6	COM7
静态显示模式	↑→	↑→	↑→	↑→	注 2	注 2	注 2	注 2
2分时模式[注 1]	↑	→	开路	开路	注 2	注 2	注 2	注 2
3分时模式[注 1]	↑		→	开路	注 2	注 2	注 2	注 2
4分时模式[注 1]	↑			→	注 2	注 2	注 2	注 2
8分时模式[注 1]	↑							→

注　**1.** 使用 segment 键扫描输出 功能（KSON = 1），在一个 LCD 输出周期之后，无显示输出将会执行一个时间片的时期。

　　2. 引脚悬空或作为 segment 引脚。

（2） Segment 信号

（a） μPD78F044x，78F045x

在 LCD 显示期间，对应 32 字节 LCD 显示数据存储器（FA40H 至 FA5FH）的 segment 信号，每个字节的第 0 位，第 1 位，第 2 位和第 3 位分别随着 COM0，COM1，COM2 和 COM3 同步读出。如果某位为 1，则其被转换为选择电压，如果某位为 0，则其被转换为非选择电压。转换结果被输出到 segment（SEG0 至 SEG31）。

而且，在 segment 键扫描输出期间，对应于端口寄存器 14 和 15 的 segment 信号。各个端口寄存器的第 0 位至第 3 位和第 4 位至第 7 位将分别在 segment 键扫描输出时期的前半部和后半部同步读取。如果各位的内容是 1 或 0，将分别输出高电平或低电平到 segment 引脚（SEG16 至 SEG23）。

（b） μPD78F046x

在 LCD 显示期间，对应 24 字节 LCD 显示数据存储器（FA40H 至 FA57H）的 segment 信号，每个字节的第 0 位，第 1 位，第 2 位和第 3 位分别随着 COM0，COM1，COM2 和 COM3 同步读出。如果某位为 1，则其被转换为选择电压，如果某位为 0，则其被转换为非选择电压。转换结果被输出到 segment（SEG0 至 SEG23）。

而且，在 segment 键扫描输出期间，对应于端口寄存器 14 和 15 的 segment 信号。各个端口寄存器的第 0 位至第 3 位和第 4 位至第 7 位将分别在 segment 键扫描输出时期的前半部和后半部同步读取。如果各位的内容是 1 或 0，将分别输出高电平或低电平到 segment 引脚（SEG16 至 SEG23）。

利用上述给定信息，可以检查 LCD 显示数据存储器内前表电极（对应于 segment 信号）和后表电极（对应于公共端信号）如何组合形成的显示模式，并以一对一的原则写入对应于期望显示模式的位数据。

在 LCD 静态显示、2 分时和 3 分时模式下，LCD 显示数据存储器的第 1 位至第 3 位、第 2 位和第 3 位、第 3 位不用于 LCD 显示。所以这些位可以用于显示以外的目的。

FA40H 至 FA43H 的高四位恒为 0。

（3） 在 LCD 显示信号输出期间，公共端信号和 segment 信号的输出波形

在 LCD 显示信号输出期间，输出到公共端信号和 segment 信号的电压如表 17-4 所示。

当公共端和 segment 信号都是选择电压时，获得±V_{LCD}的显示打开电压。信号的其他组合情况对应显示关闭电压。

表 17-4. LCD 驱动电压

（a） 静态显示模式（在 LCD 显示信号输出期间）

公共端信号 \ Segment 信号	选择信号电平	非选择信号电平
	V_{SS}/V_{LC0}	V_{LC0}/V_{SS}
V_{LC0}/V_{SS}	$-V_{LCD}/+V_{LCD}$	0 V/0 V

（b） 1/2 偏压方式（在 LCD 显示信号输出期间）

公共端信号 \ Segment 信号		选择信号电平	非选择信号电平
		V_{SS}/V_{LC0}	V_{LC0}/V_{SS}
选择信号电平	V_{LC0}/V_{SS}	$-V_{LCD}/+V_{LCD}$	0 V/0 V
非选择信号电平	$V_{LC1} = V_{LC2}$	$-\frac{1}{2}V_{LCD}/+\frac{1}{2}V_{LCD}$	$+\frac{1}{2}V_{LCD}/-\frac{1}{2}V_{LCD}$

（c） 1/3 偏压方式（在 LCD 显示信号输出期间）

公共端信号 \ Segment 信号		选择信号电平	非选择信号电平
		V_{SS}/V_{LC0}	V_{LC1}/V_{LC2}
选择信号电平	V_{LC0}/V_{SS}	$-V_{LCD}/+V_{LCD}$	$-\frac{1}{3}V_{LCD}/+\frac{1}{3}V_{LCD}$
非选择信号电平	V_{LC2}/V_{LC1}	$-\frac{1}{3}V_{LCD}/+\frac{1}{3}V_{LCD}$	$+\frac{1}{3}V_{LCD}/-\frac{1}{3}V_{LCD}$

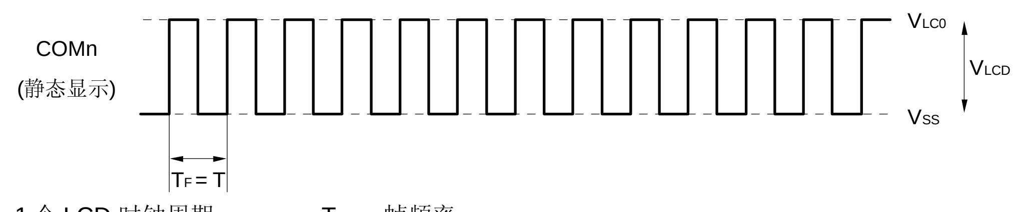
（d） 1/4 偏压方式（在 LCD 显示信号输出期间）

公共端信号 \ Segment 信号		选择信号电平	非选择信号电平
		V_{LC0}/V_{SS}	V_{LC1}/V_{LC2}
选择信号电平	V_{SS}/V_{LC0}	$+V_{LCD}/-V_{LCD}$	$+\frac{1}{2}V_{LCD}/-\frac{1}{2}V_{LCD}$
非选择信号电平	V_{LC1}/V_{LC3}	$+\frac{1}{4}V_{LCD}/-\frac{1}{4}V_{LCD}$	$-\frac{1}{4}V_{LCD}/+\frac{1}{4}V_{LCD}$

图 17-13 展示了公共端信号波形， 图 17-14 展示了公共端和 segment 信号的电压和相位。

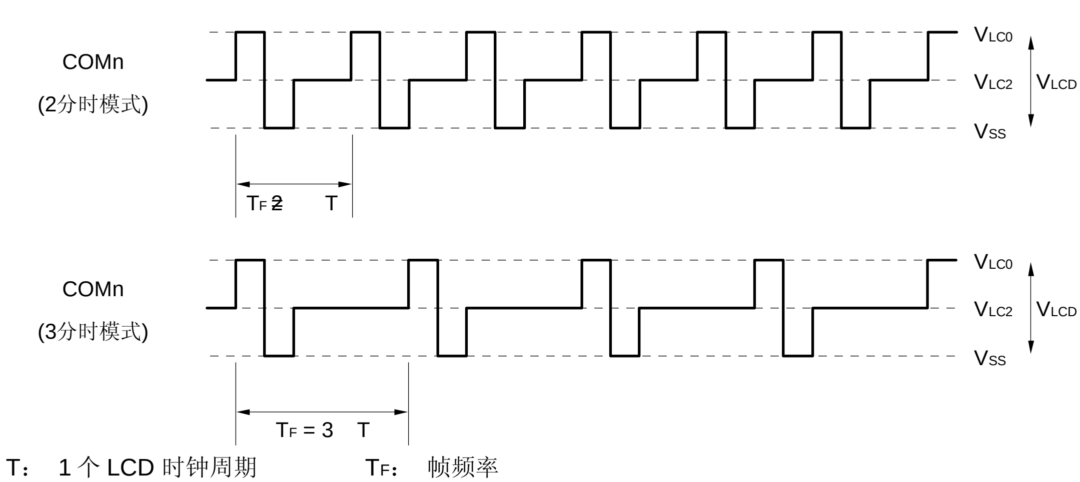
图 **17-13.** 公共端信号波形

（**a**） 静态显示模式



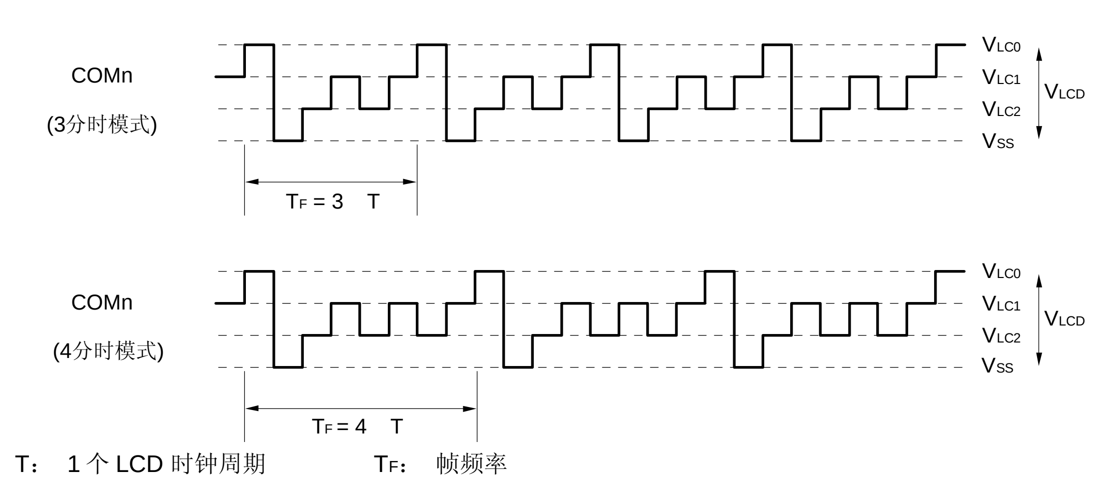
T： 1 个 LCD 时钟周期 TF： 帧频率

（**b**） **1/2** 偏压方式



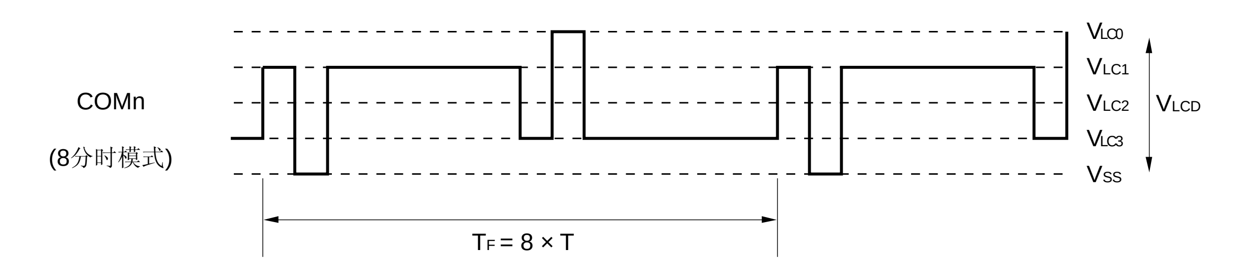
T： 1 个 LCD 时钟周期 TF： 帧频率

（**c**） **1/3** 偏压方式



T： 1 个 LCD 时钟周期 TF： 帧频率

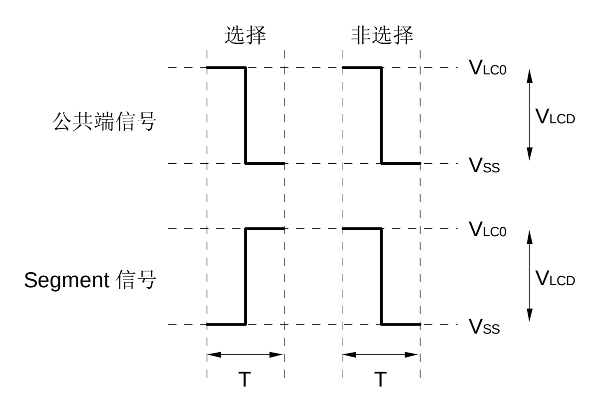
（d）　1/4 偏压方式



T：　1 个 LCD 时钟周期　　　　TF：　帧频率

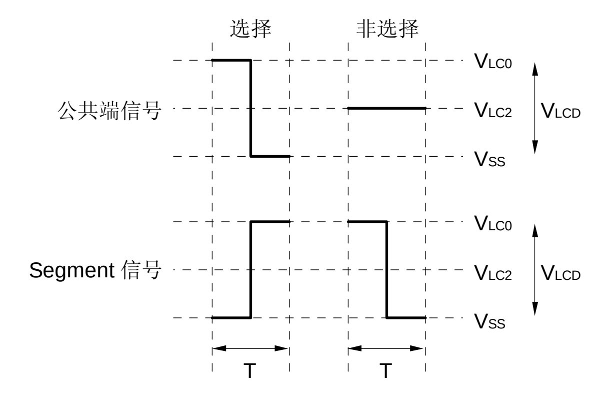
图 **17-14.** 公共端和 **Segment** 信号的电压和相位

（**a**）　静态显示模式



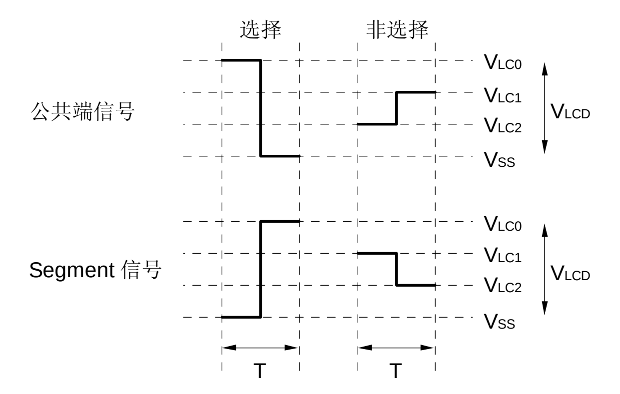
T：　1 个 LCD 时钟周期

（**b**）　**1/2** 偏压方式



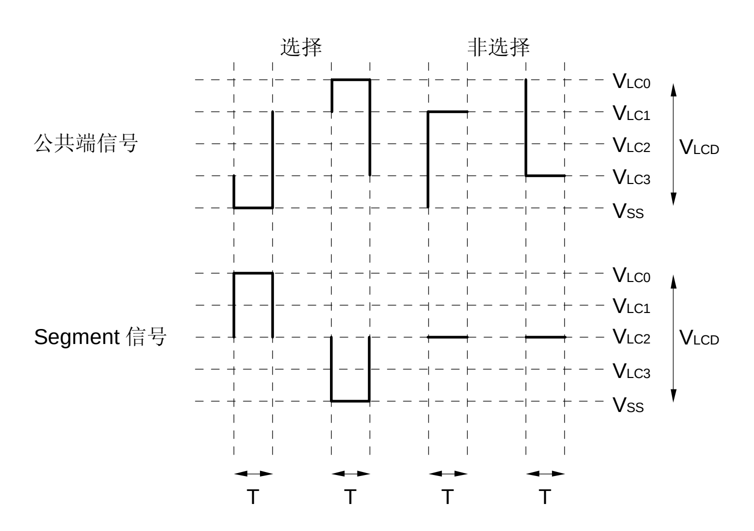
T：　1 个 LCD 时钟周期

（**c**）　**1/3** 偏压方式



T：　1 个 LCD 时钟周期

（d）　1/4 偏压方式



T：　1 个 LCD 时钟周期

（4）　在 segment 键扫描输出期间，公共端和 segment 信号的输出波形

在 segment 键扫描输出期间，输出到公共端信号和 segment 信号的电压如表 17-5 所示。

当公共端和 segment 信号都是选择电压时，获得$\pm V_{LCD}$的显示打开电压。信号的其他组合情况对应显示关闭电压。

表 17-5. LCD 驱动电压

（a）　1/2 偏压方式（在 segment 键扫描输出期间）

键扫描信号 公共端信号		P14x = P15x = 1	P14x = P15x = 0
		V_{D0}	V_{SS}
非选择信号电平	$V_{LC1} = V_{LC2}$	$+\frac{1}{2}V_{LCD}$	$-\frac{1}{2}V_{LCD}$

（b）　1/3 偏压方式（在 segment 键扫描输出期间）

键扫描信号 公共端信号		P14x = P15x = 1	P14x = P15x = 0
		V_{DD}	V_{SS}
非选择信号电平	V_{LC2}/V_{LC1}	$+\frac{2}{3}V_{LCD}/+\frac{1}{3}V_{LCD}$	$-\frac{1}{3}V_{LCD}/-\frac{2}{3}V_{LCD}$

（c）　1/4 偏压方式（在 segment 键扫描输出期间）

键扫描信号 公共端信号		P14x = P15x = 1	P14x = P15x = 0
		V_{DD}	V_{SS}
非选择信号电平	V_{LC1}/V_{LC3}	$+\frac{1}{4}V_{LCD}/+\frac{3}{4}V_{LCD}$	$-\frac{3}{4}V_{LCD}/-\frac{1}{4}V_{LCD}$

备注　静态显示模式下，不能使用 segment 键扫描输出功能。

图 17-15 展示了公共端信号波形，图 17-16 展示了公共端和 segment 信号的电压和相位。

图 17-15 公共端信号波形

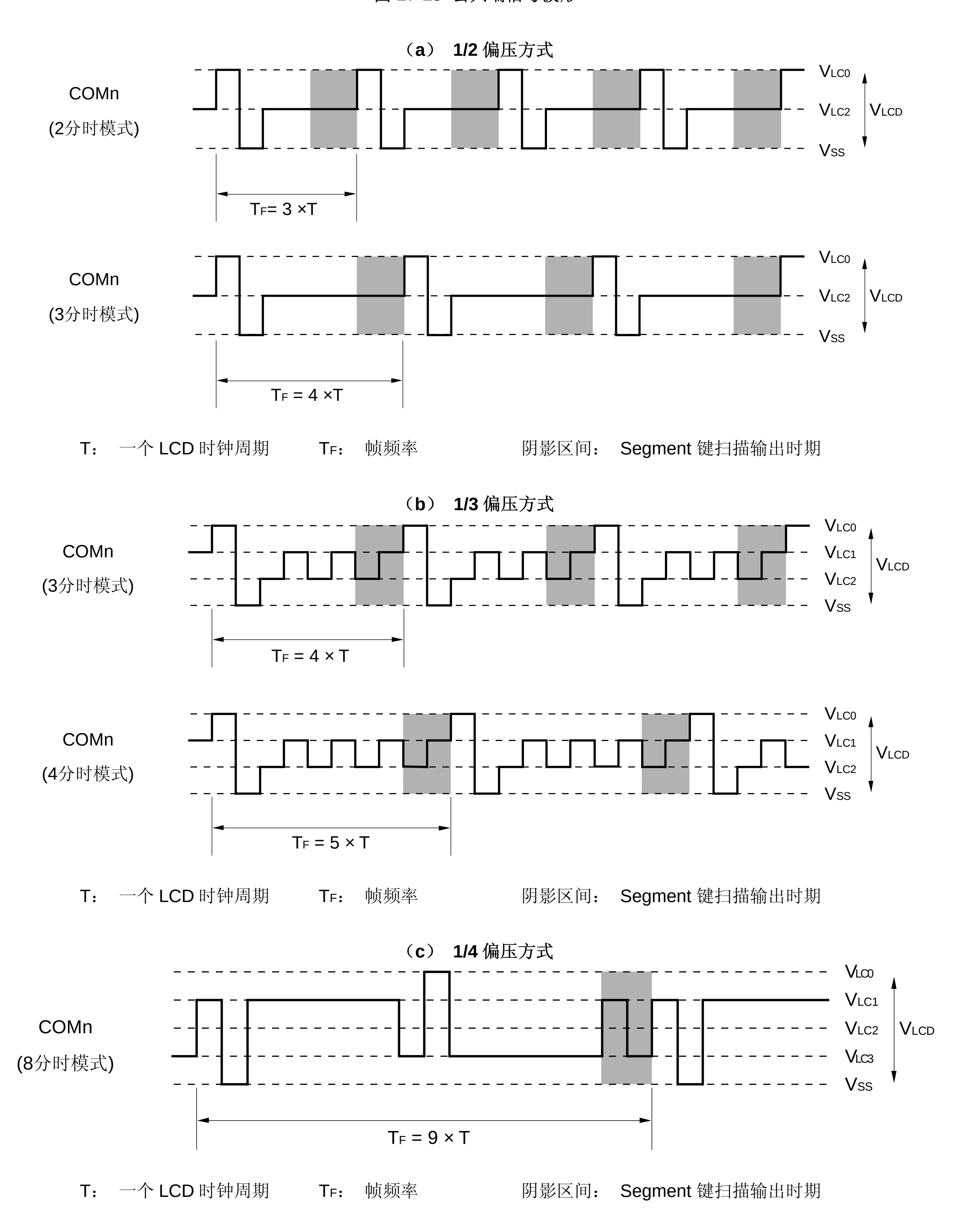
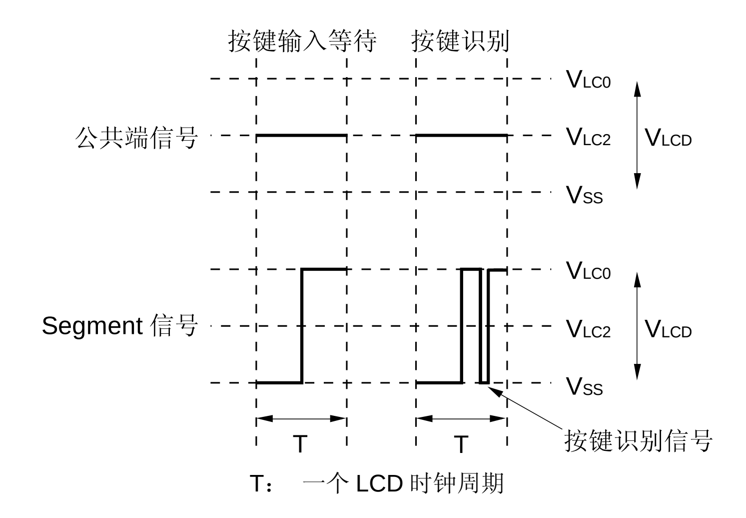


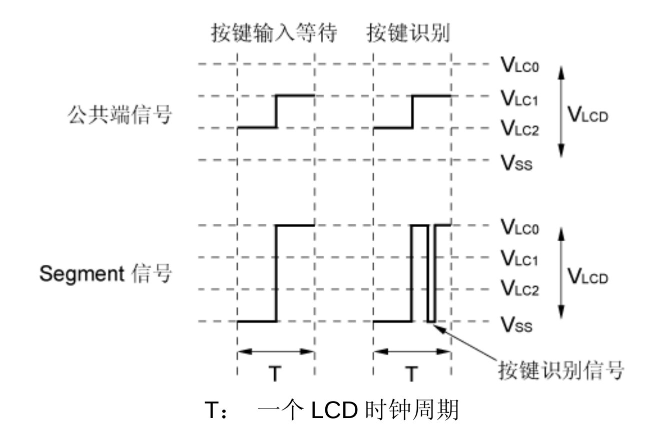
图 17-16　公共端和 Segment 信号的电压和相位

（a）　1/2 偏压方式



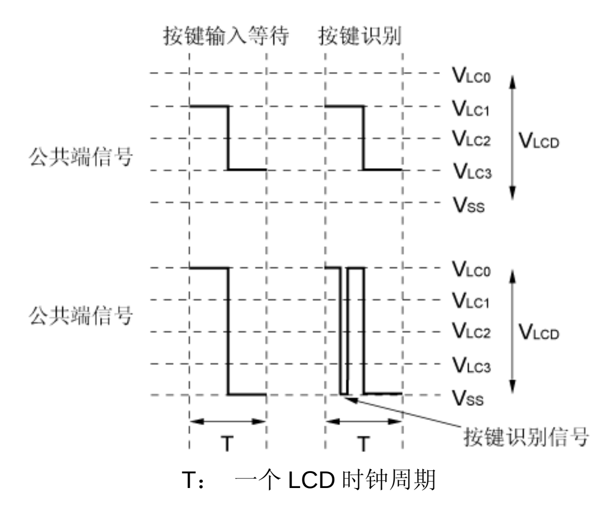
T：　一个 LCD 时钟周期

（b）　1/3 偏压方式



T：　一个 LCD 时钟周期

（c）　1/4 偏压方式



T：　一个 LCD 时钟周期

备注　　Segment 键扫描信号必须通过端口寄存器 14 和 15（P14，P15）来设置。

17.7 显示模式

17.7.1 静态显示示例

图 17-18 展示了如何把 3 个数字 LCD 面板连接到 78K0/LE3 芯片的 segment 信号（SEG0 至 SEG23）和公共端信号（COM0），该面板具有的显示模式如图 17-17 所示。这个示例在 LCD 面板上显示数字"12.3" 。显示数据存储器（FA40H 至 FA57H 地址）的内容对应该显示。

以下描述重点关注显示在第二个字段的数字"2" （☐）。为了在 LCD 面板显示"2" ，必需在公共端信号 COM0 的时序，按照表 17-6 所示将选择电压或非选择电压输出到 SEG8 至 SEG15 引脚；segment 信号和 LCD 段的关系，参见图 **17-17** 。

表 17-6. 选择和非选择电压 （COM0）

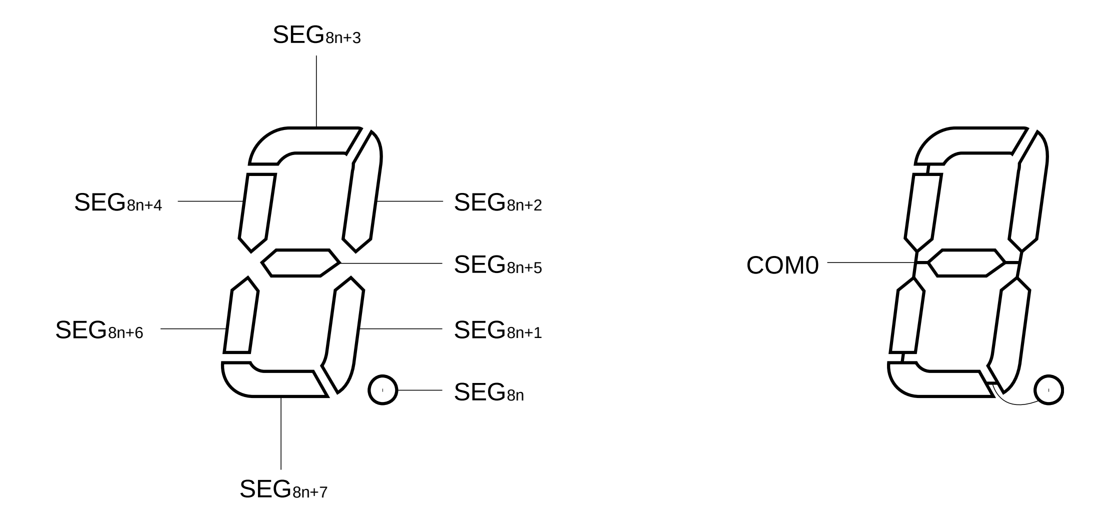
Segment / 公共端	SEG8	SEG9	SEG10	SEG11	SEG12	SEG13	SEG14	SEG15
COM0	选择	非选择	选择	选择	非选择	选择	选择	选择

根据表 17-6，它决定显示数据存储器区域（FA48H 至 FA4FH）的位 0 模型必须是 10110111。

图 17-19 展示了 SEG11 和 SEG12 以及 COM0 的 LCD 驱动波形。在 COM0 的时序选择电压输出到 SEG11 时，产生一个交替矩形波形，$+V_{LCD}/-V_{LCD}$ ，来打开对应的 LCD 段。

COM1 至 COM3 的波形和 COM0 相同。所以，COM0 至 COM3 可以连接在一起，增加驱动能力。

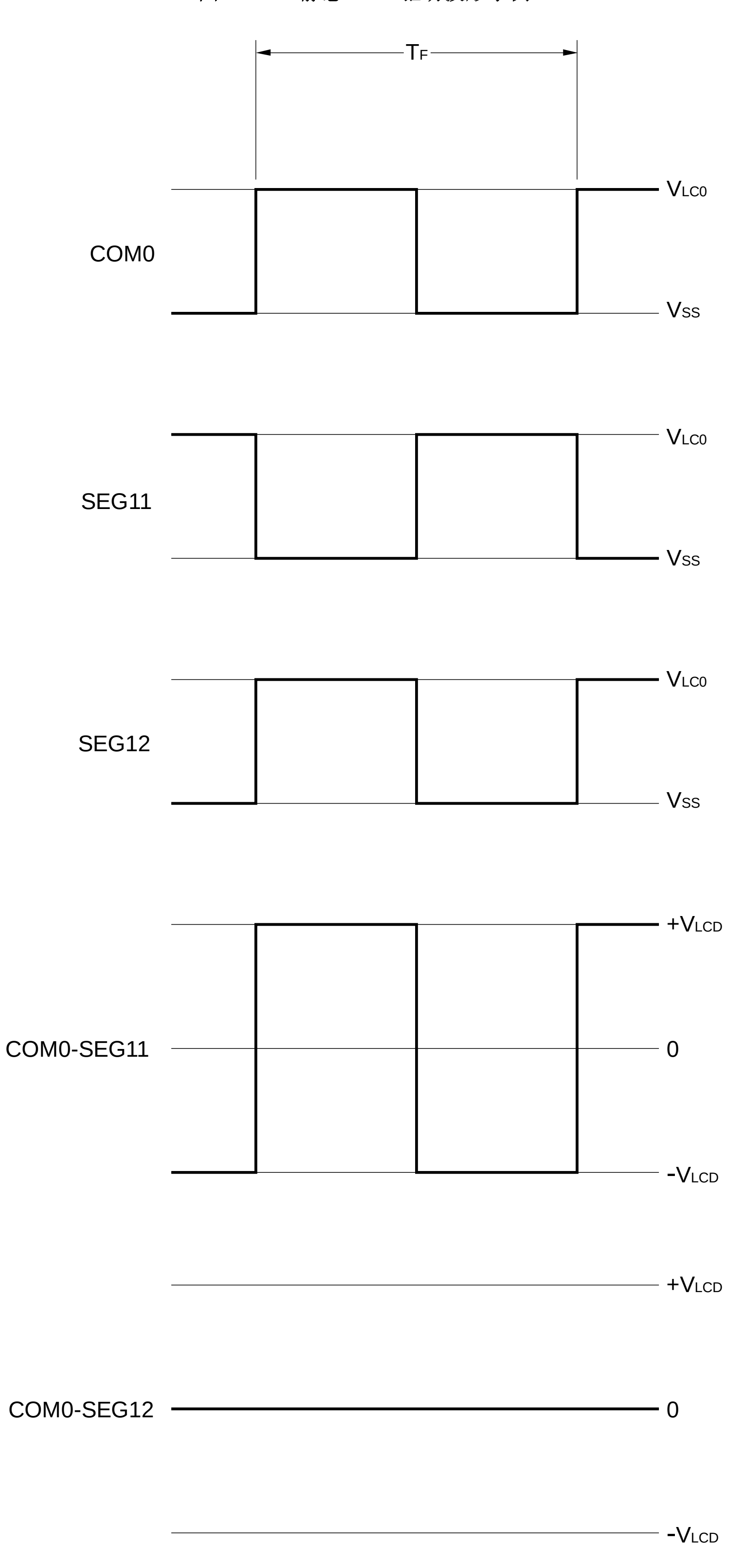
图 17-17. 静态 LCD 显示模型和电极连接



备注　　n = 0 至 2

图 17-18. 静态 LCD 面板的连接示例

图 17-19. 静态 LCD 驱动波形示例



17.7.2 2 分时显示示例

图 17-21 展示了如何把 6 个数字 LCD 面板连接到 78K0/LE3 芯片的 segment 信号（SEG0 至 SEG23）和公共端信号（COM0 和 COM1），该面板具有的显示模式如图 17-20 所示。这个示例在 LCD 面板上显示数字"12345.6"。显示数据存储器（FA40H 至 FA57H 地址）的内容对应该显示。

以下描述重点关注显示在第四个字段的数字"3"（□）。为了在 LCD 面板显示"3"，必需在公共端信号 COM0 和 COM1 的时序，按照表 17-7 所示将选择电压或非选择电压输出到 SEG12 至 SEG15 引脚；segment 信号和 LCD 段的关系，参见图 **17-20**。

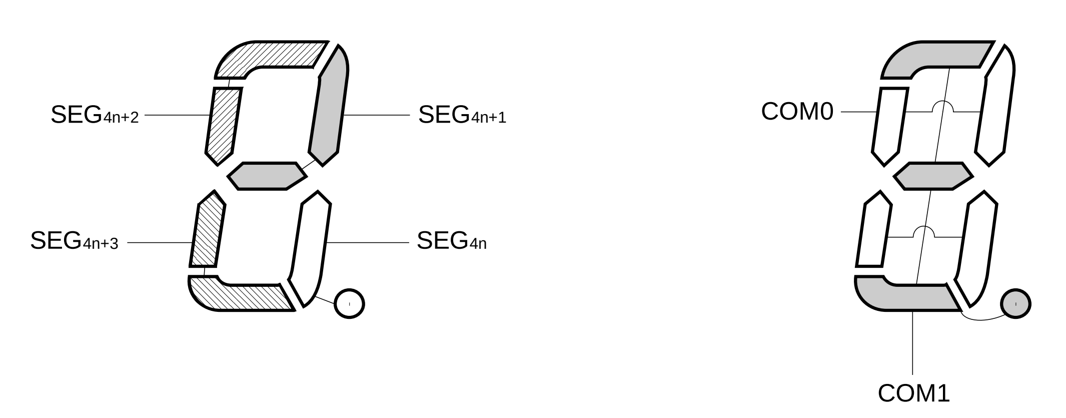
表 17-7. 选择和非选择电压（COM0 和 COM1）

Segment / 公共端	SEG12	SEG13	SEG14	SEG15
COM0	选择	选择	非选择	非选择
COM1	非选择	选择	选择	选择

根据表 17-7，它决定对应 SEG15 的显示数据存储器区域（FA4FH）必须包含 xx10。

图 17-22 展示了 SEG15 和各个公共端信号之间的 LCD 驱动波形示例。在 COM1 的时序选择电压输出到 SEG15 时，产生一个交替矩形波形，$+V_{LCD}/-V_{LCD}$，来打开对应的 LCD 段。

图 17-20. 2 分时 LCD 显示模型和电极连接



备注 n = 0 至 5

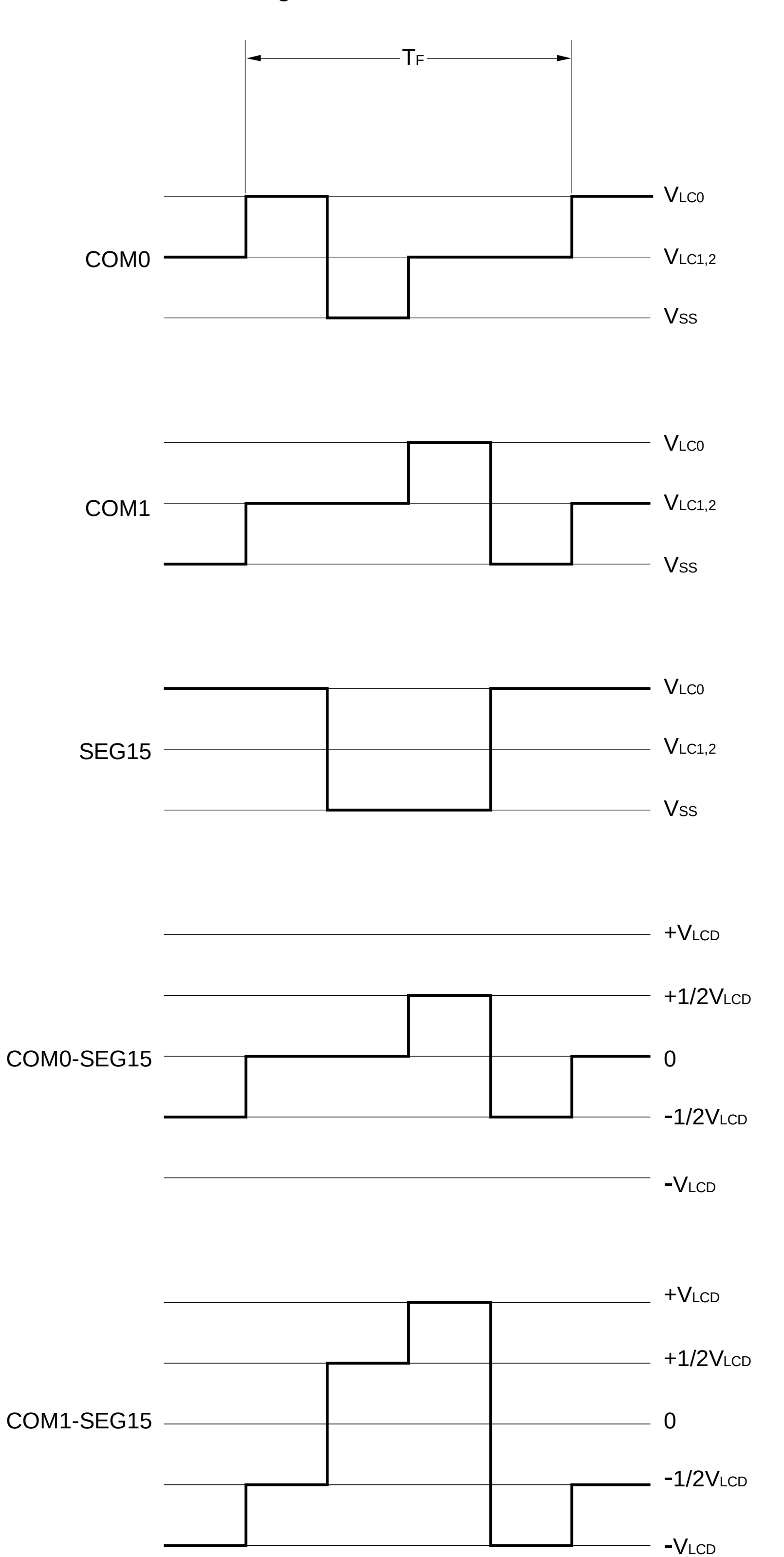
图 17-21. 2 分时 LCD 面板的连接示例

地址	第3位	第2位	第1位	第0位
FA40H	×	×	0	1
1	×	×	1	0
2	×	×	1	1
3	×	×	1	1
4	×	×	1	1
5	×	×	1	0
6	×	×	1	1
7	×	×	1	0
8	×	×	0	1
9	×	×	1	1
A	×	×	0	1
B	×	×	0	0
C	×	×	0	1
D	×	×	1	1
E	×	×	1	0
F	×	×	1	0
FA50H	×	×	0	0
1	×	×	1	1
2	×	×	1	0
3	×	×	1	1
4	×	×	0	1
5	×	×	0	1
6	×	×	0	0
7	×	×	0	0

×： 因为使用 2 分时模式，总是可以用来存储任意数据。

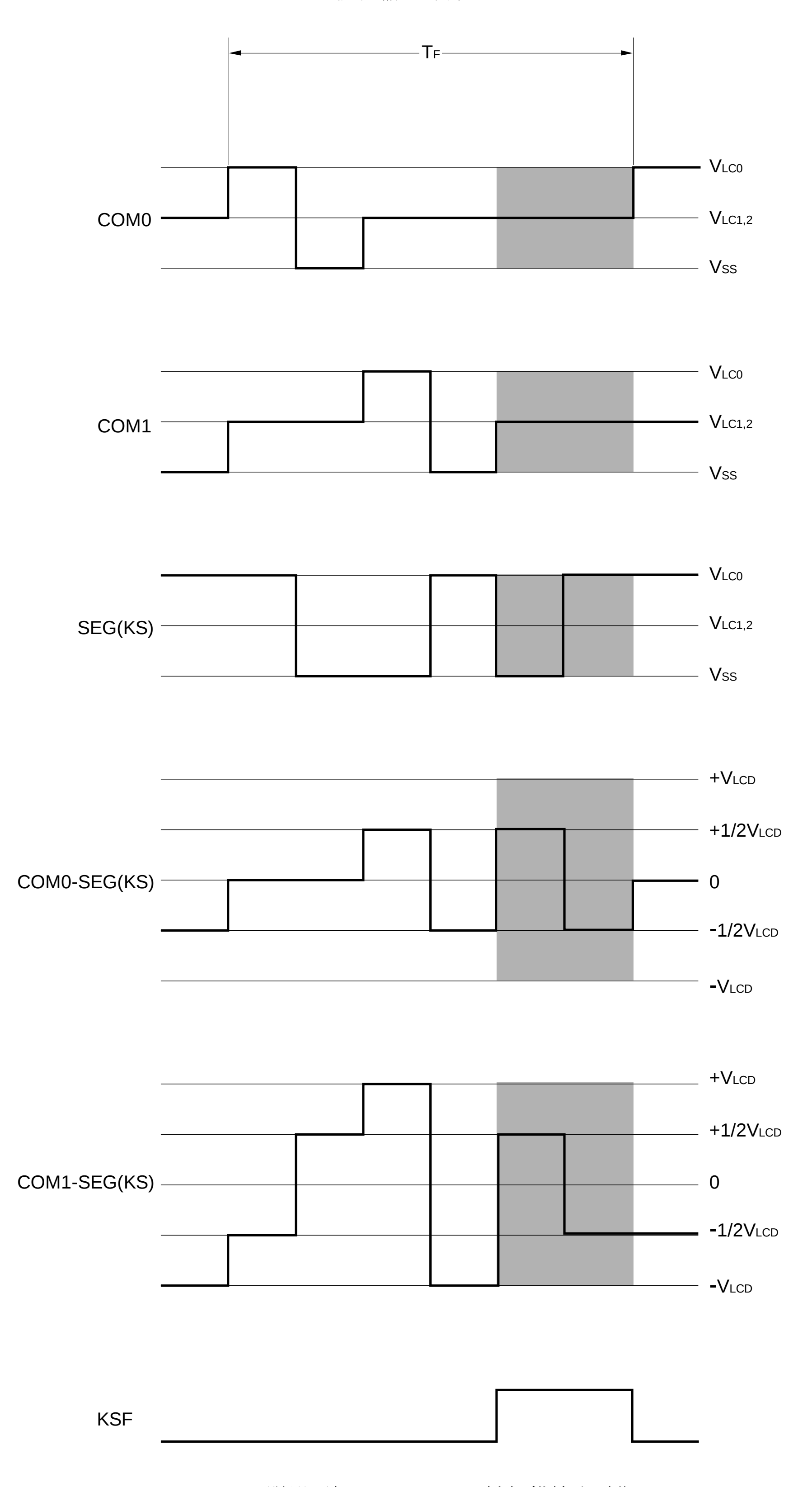
图 **17-22. 2** 分时 **LCD** 驱动波形示例（**1/2** 偏压方式）

（**a**） 未使用 **segment** 键扫描功能时（**KSON = 0**）



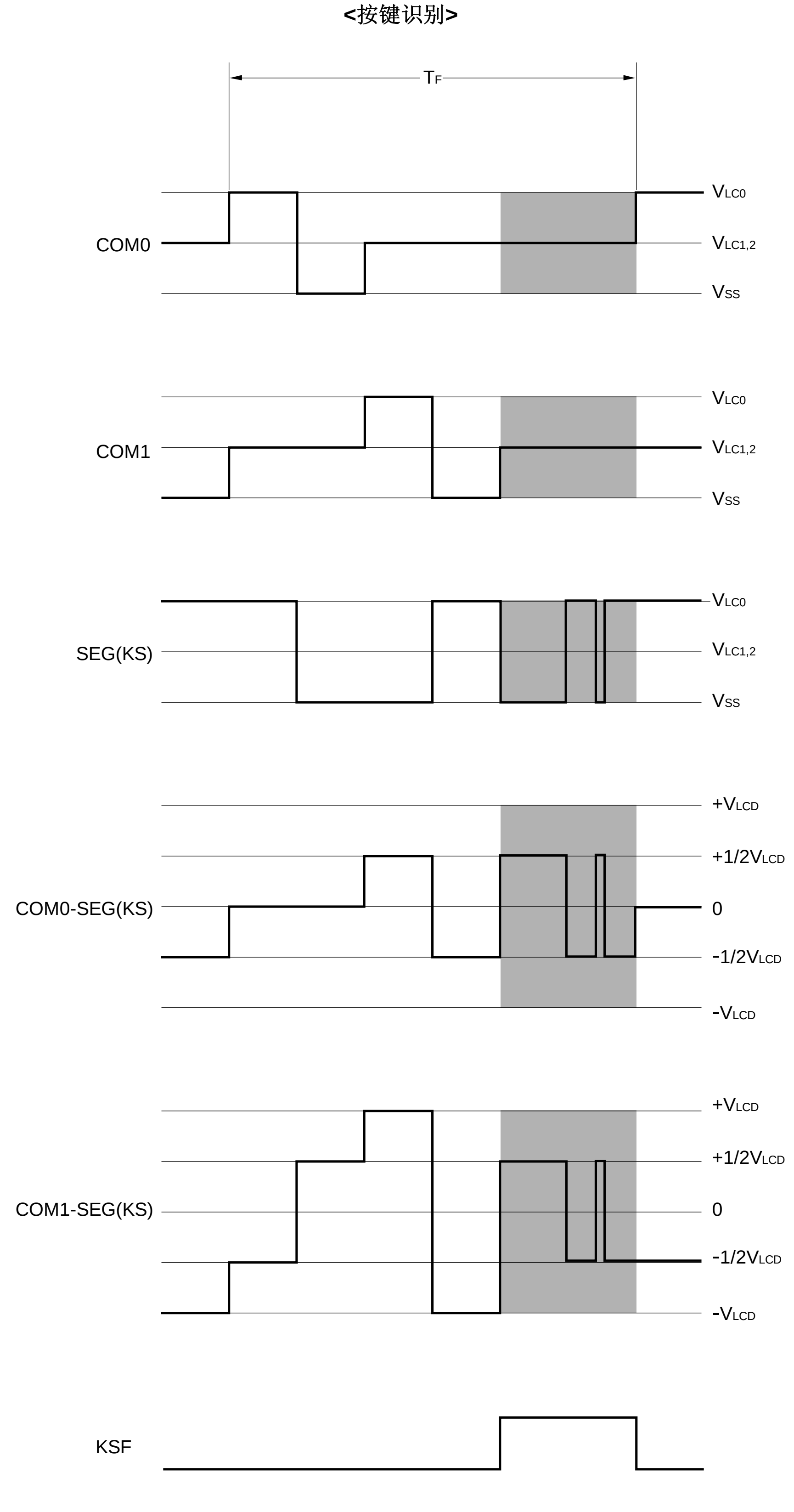
（**b**） 使用键扫描功能时（**KSON = 1**）

<按键输入等待>



阴影区间：　**Segment** 键扫描输出时期

<按键识别>



阴影区间：　**Segment** 键扫描输出时期

备注　在按键识别期间，通过输出一个前半部和后半部具有反向关系的信号，可以消除 LCD 面板的剩余电荷。

17.7.3 3 分时显示示例

图 17-24 展示了如何把 8 个数字 LCD 面板连接到 78K0/LE3 芯片的 segment 信号（SEG0 至 SEG23）和公共端信号（COM0 至 COM2），该面板具有的显示模式如图 17-23 所示。这个示例在 LCD 面板上显示数字"1234567.8"。显示数据存储器（FA40H 至 FA57H 地址）的内容对应该显示。

以下描述重点关注显示在第三个字段的数字"6"（□）。为了在 LCD 面板显示"6"，必需在公共端信号 COM0 至 COM2 的时序，按照表 17-8 所示将选择电压或非选择电压输出到 SEG6 至 SEG8 引脚；segment 信号和 LCD 段的关系，参见图 **17-23**。

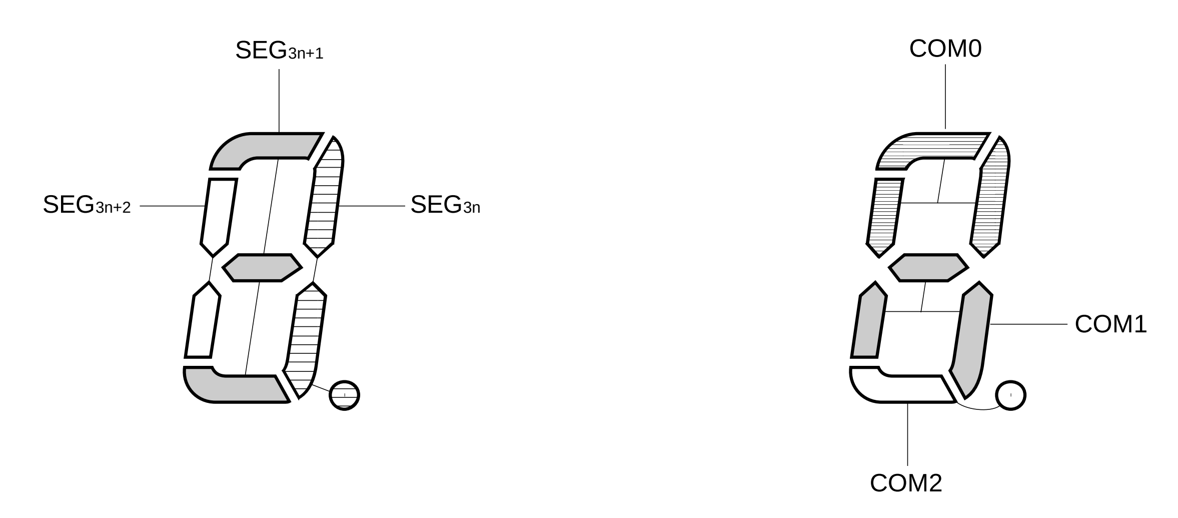
表 17-8. 选择和非选择电压 （COM0 至 COM2）

公共端 \ Segment	SEG6	SEG7	SEG8
COM0	非选择	选择	选择
COM1	选择	选择	选择
COM2	选择	选择	–

根据表 17-8，它决定对应 SEG6 的显示数据存储器区域（FA46H）必须包含 x110。

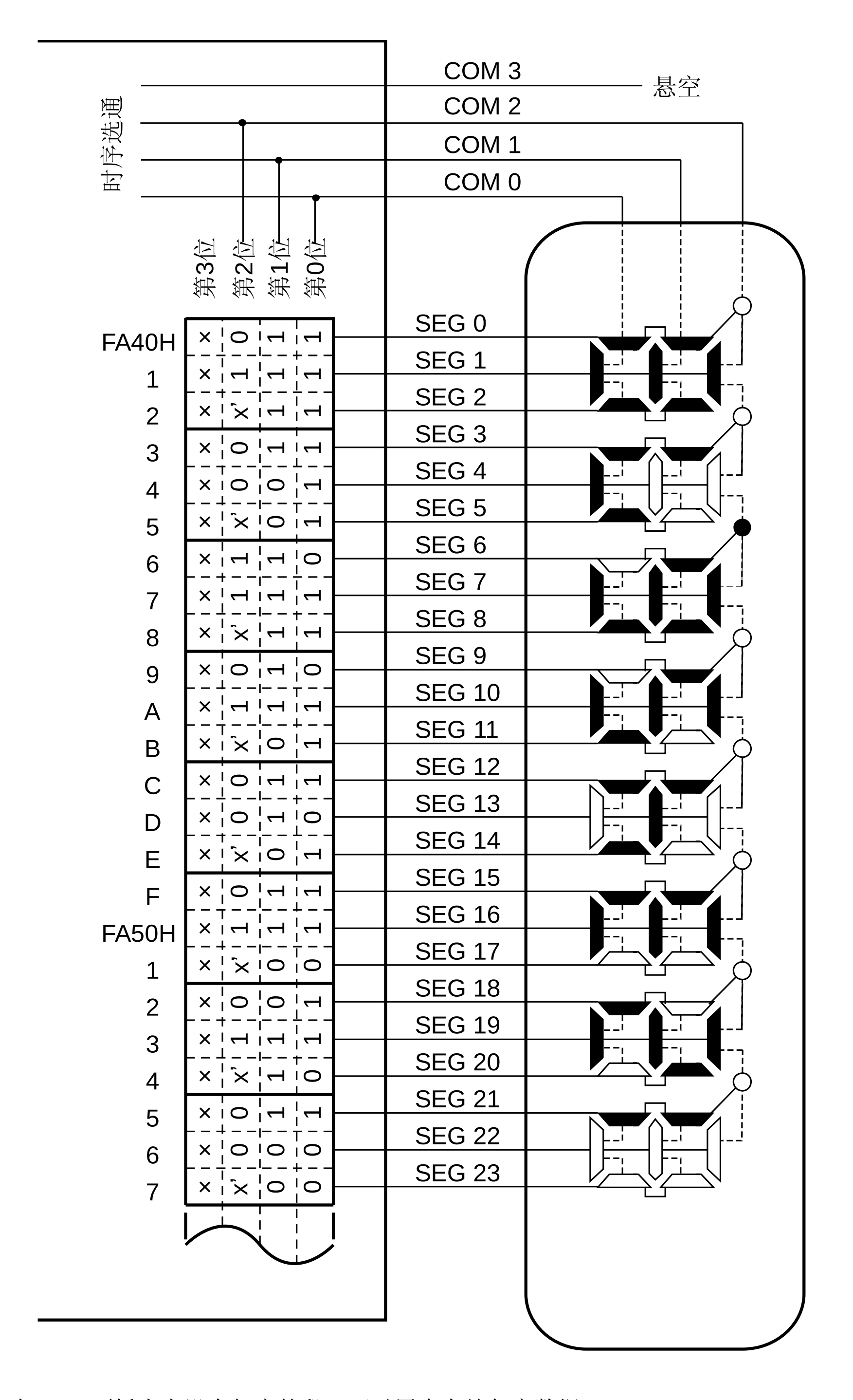
图 17-25 和图 17-26 展示了在 1/2 和 1/3 偏压模式下，SEG6 和各个公共端信号之间的 LCD 驱动波形示例。在 COM1 或 COM2 的时序将选择电压输出到 SEG6 时，产生一个交替矩形波形，$+V_{LCD}/-V_{LCD}$，来打开对应的 LCD 段。

图 17-23. 3 分时 LCD 显示模型和电极连接



备注 n = 0 至 7

图 **17-24. 3** 分时 **LCD** 面板的连接示例

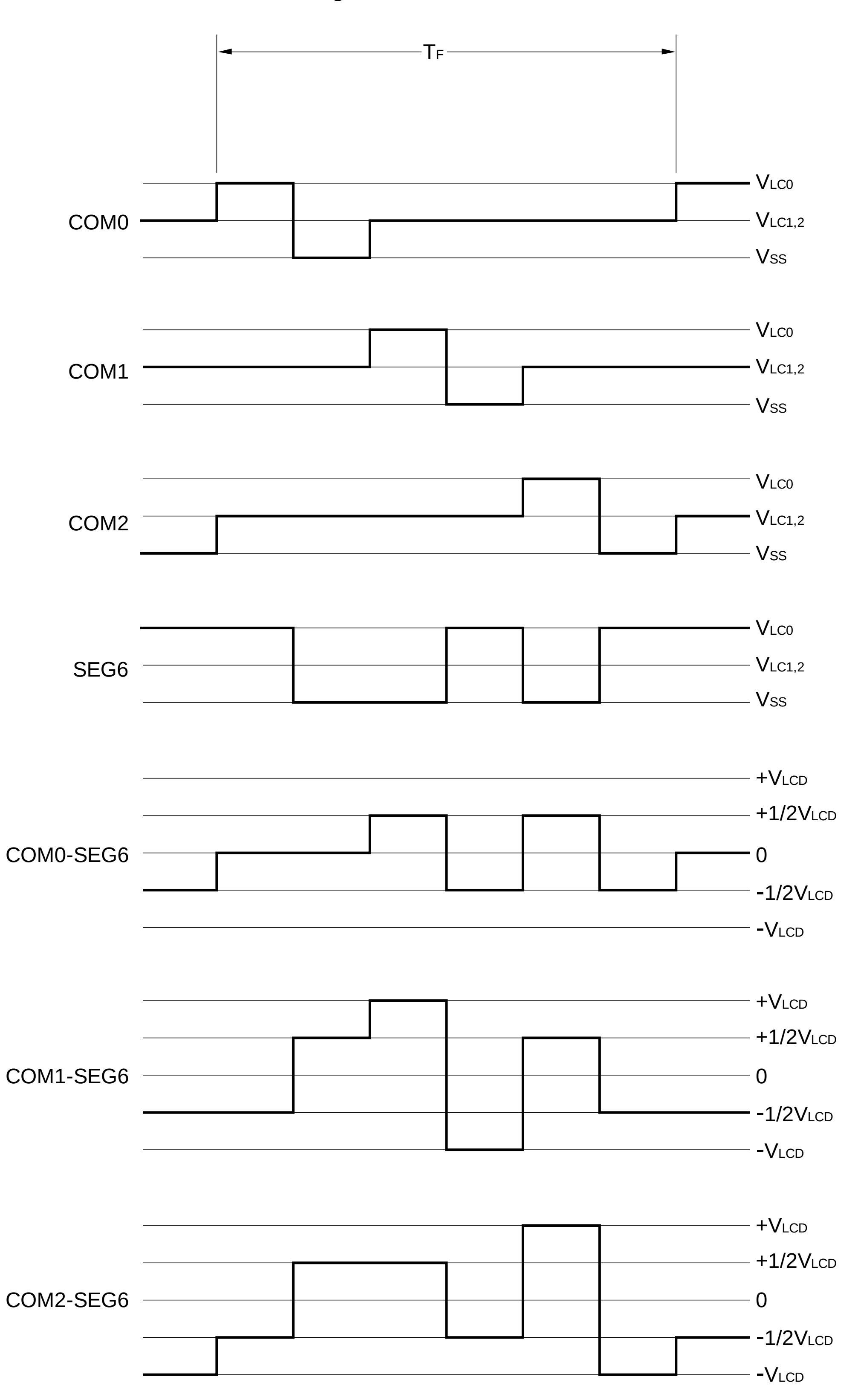


（'：　因为在 LCD 面板内上没有相应的段，可以用来存储任意数据。

×：　因为使用 3 分时模式，总是可以用来存储任意数据。

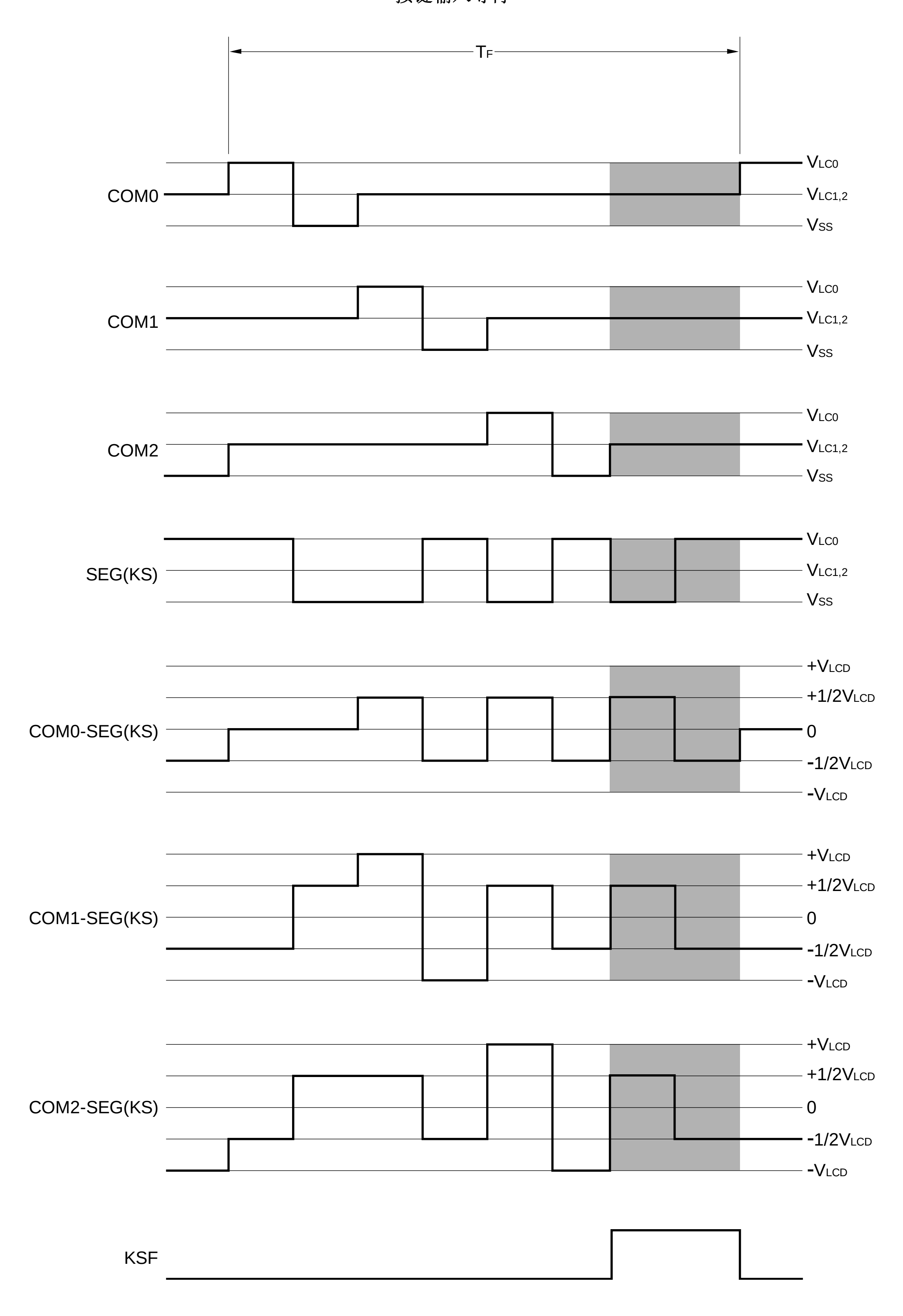
图 17-25. 3 分时 LCD 驱动波形示例（1/2 偏压方式）

（a） 未使用 segment 键扫描功能时（KSON = 0）



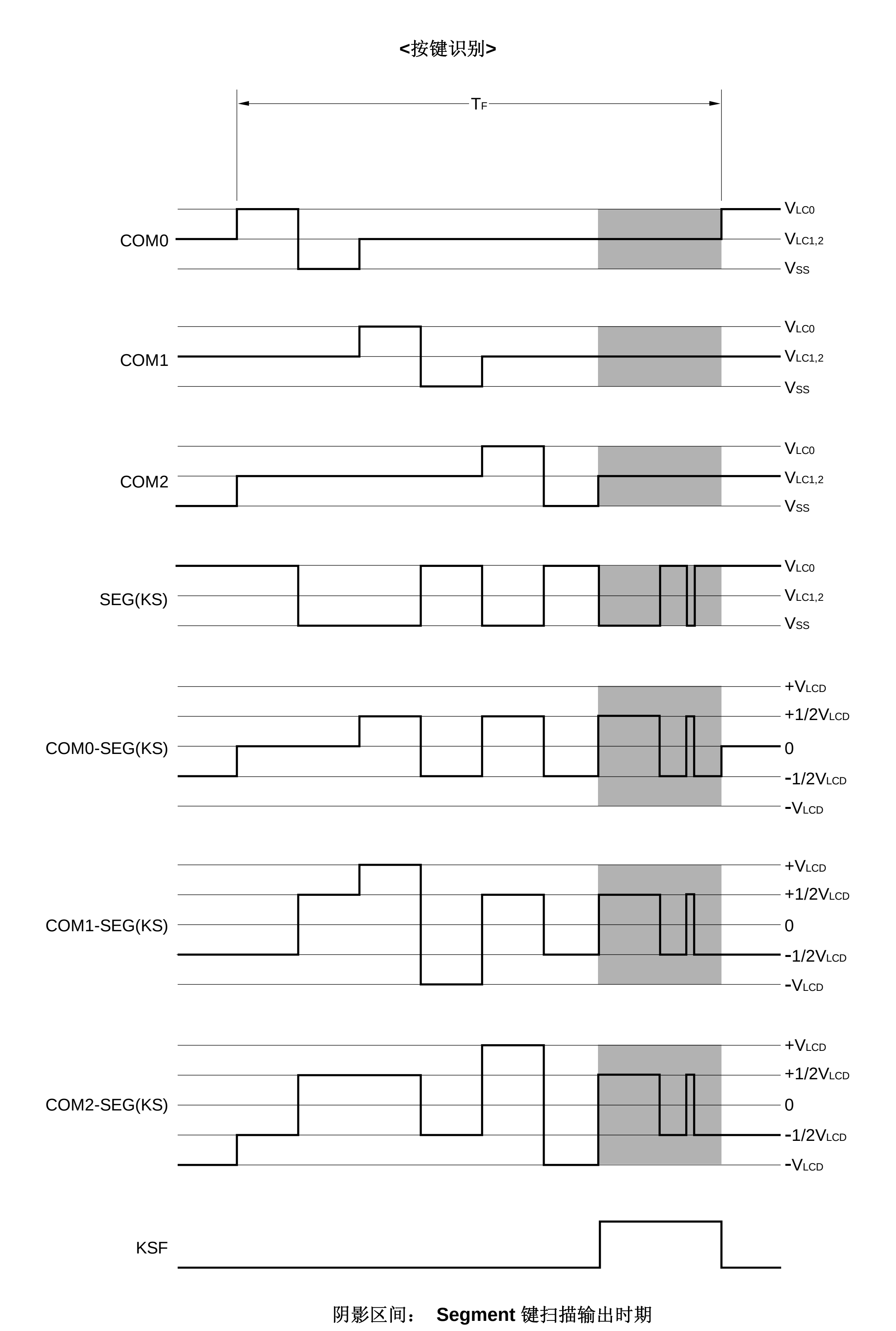
（**b**） 使用键扫描功能时（**KSON = 1**）

<按键输入等待>



阴影区间： **Segment** 键扫描输出时期

<按键识别>

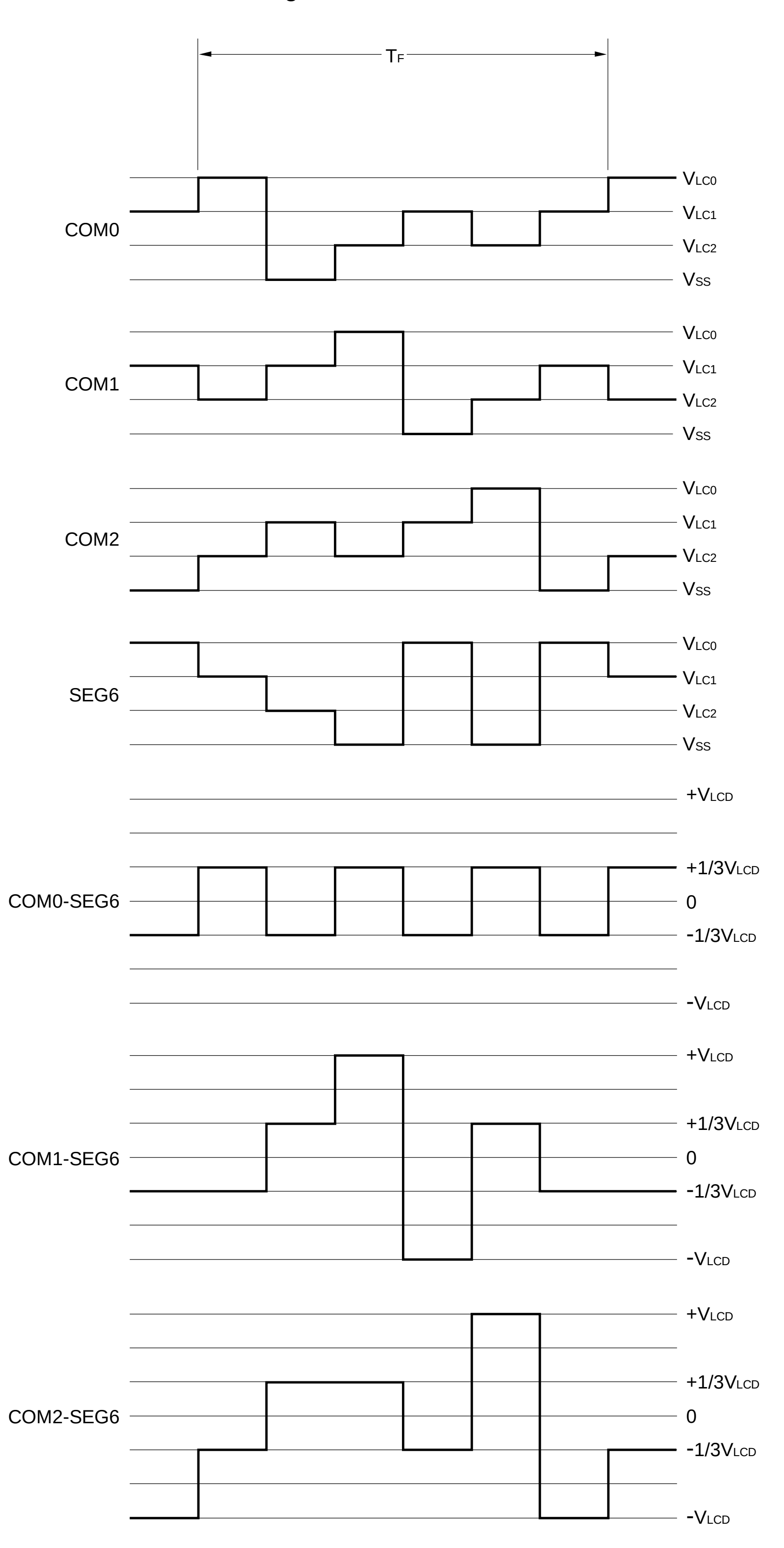


阴影区间：　**Segment 键扫描输出时期**

备注　　在按键识别期间，通过输出一个前半部和后半部具有反向关系的信号，可以消除 LCD 面板的剩余电荷。

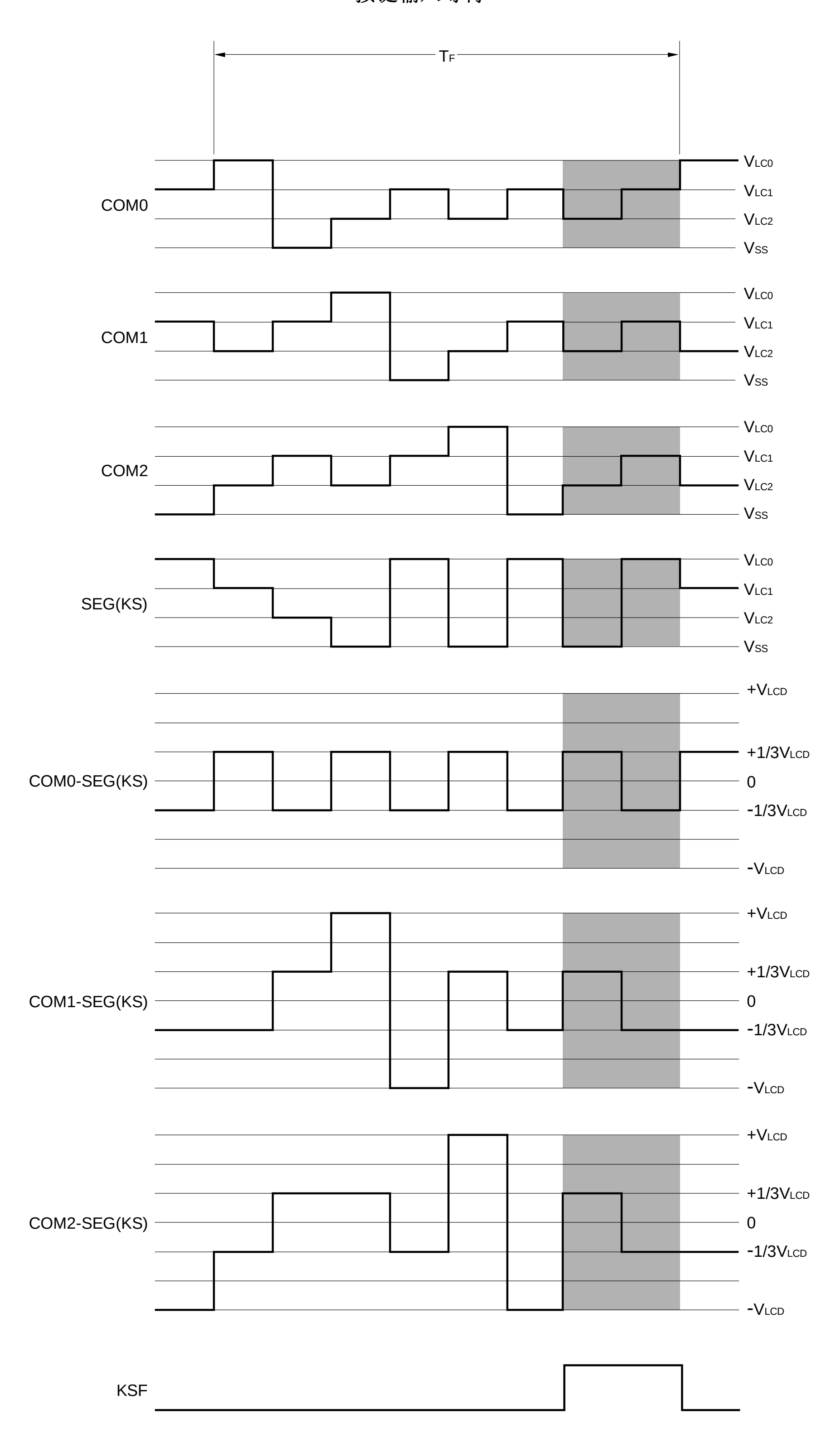
图 **17-26. 3** 分时 **LCD** 驱动波形示例（**1/3** 偏压方式）

（**a**） 未使用 **segment** 键扫描功能时（**KSON = 0**）



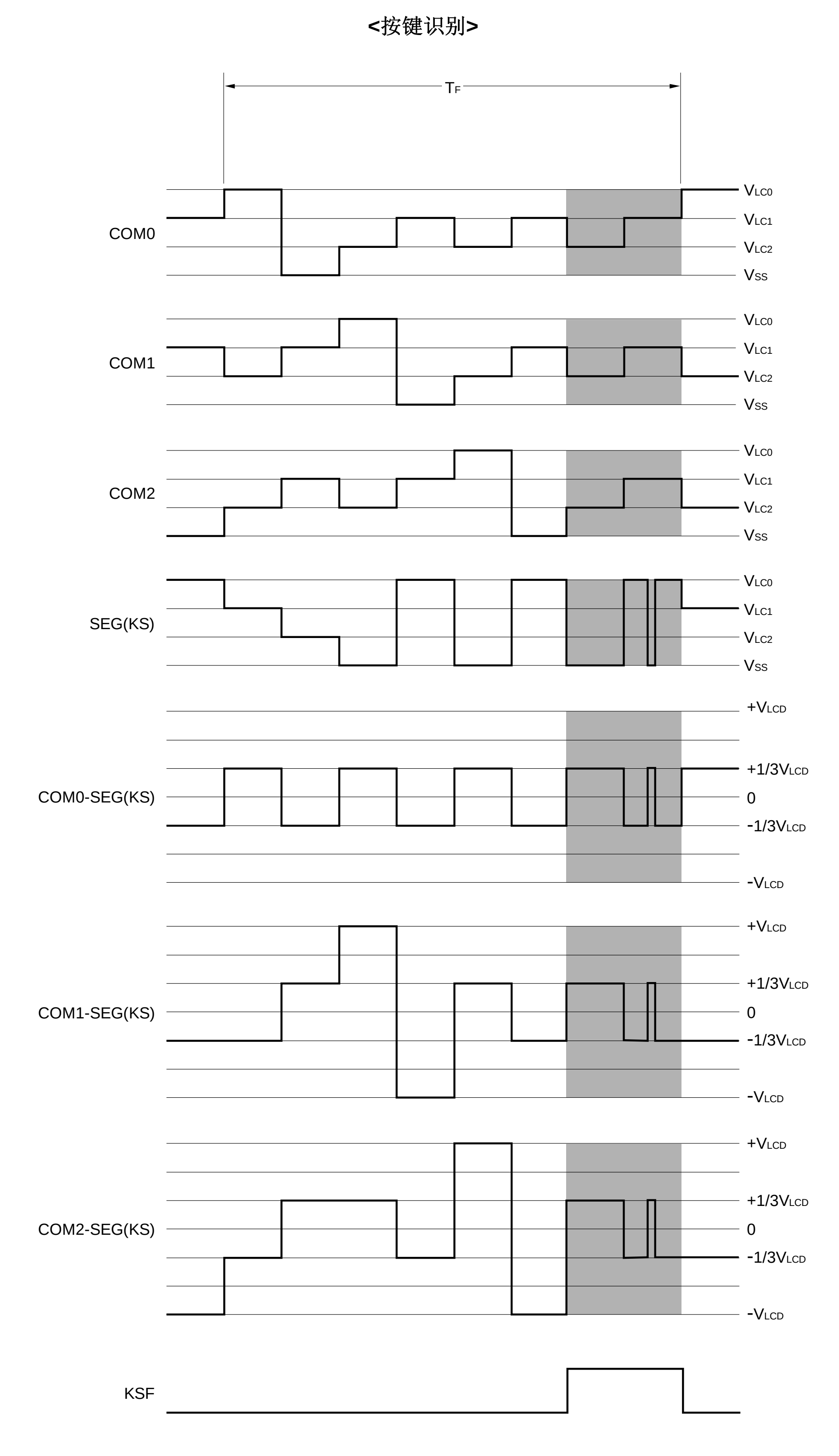
（b） 使用 segment 键扫描功能时（KSON = 1）

<按键输入等待>



阴影区间：　Segment 键扫描输出时期

<按键识别>



阴影区间：　**Segment** 键扫描输出时期

备注　　在按键识别期间，通过输出一个前半部和后半部具有反向关系的信号，可以消除 LCD 面板的剩余电荷。

17.7.4 4 分时显示示例

图 17-28 展示了如何把 12 个数字 LCD 面板连接到 78K0/LE3 芯片的 segment 信号（SEG0 至 SEG23）和公共端信号（COM0 至 COM3），该面板具有的显示模式如图 17-27 所示。这个示例在 LCD 面板上显示数字 "123456.789012"。显示数据存储器（FA40H 至 FA57H 地址）的内容对应该显示。

以下描述重点关注显示在第七个字段的数字"6"（□）。为了在 LCD 面板显示"6"，必需在公共端信号 COM0 至 COM3 的时序，按照表 17-9 所示将选择电压或非选择电压输出到 SEG12 和 SEG13 引脚；segment 信号和 LCD 段的关系，参见图 **17-27**。

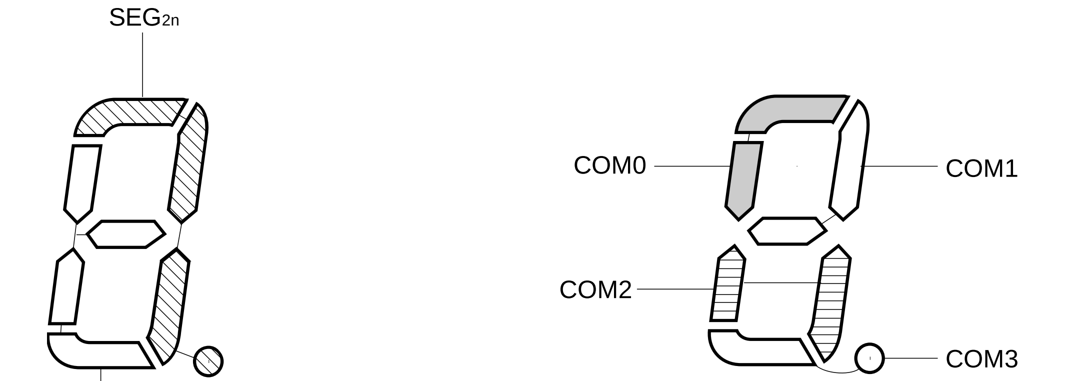
表 17-9. 选择和非选择电压（COM0 至 COM3）

公共端 \ Segment	SEG12	SEG13
COM0	选择	选择
COM1	非选择	选择
COM2	选择	选择
COM3	选择	选择

根据表 17-9，它决定对应 SEG12 的显示数据存储器区域（FA4CH）必须包含 1101。

图 17-29 展示了 SEG12 和各个公共端信号之间的 LCD 驱动波形示例。在 COM0 的时序将选择电压输出到 SEG12 时，产生一个交替矩形波形，$+V_{LCD}/-V_{LCD}$，来打开对应的 LCD 段。

图 17-27. 4 分时 LCD 显示模型和电极连接



备注 n = 0 至 11

图 17-28. 4 分时 LCD 面板的连接示例

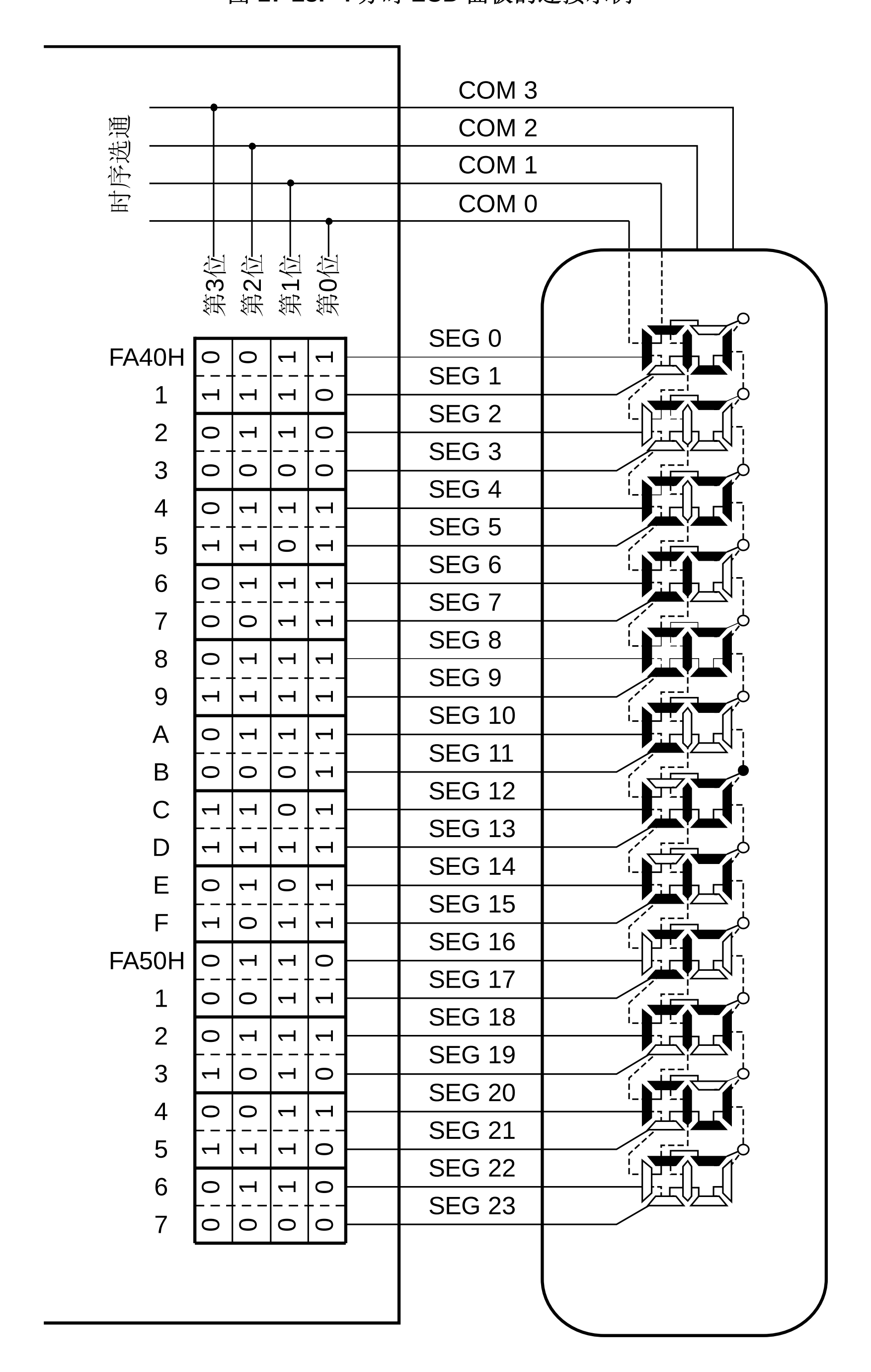
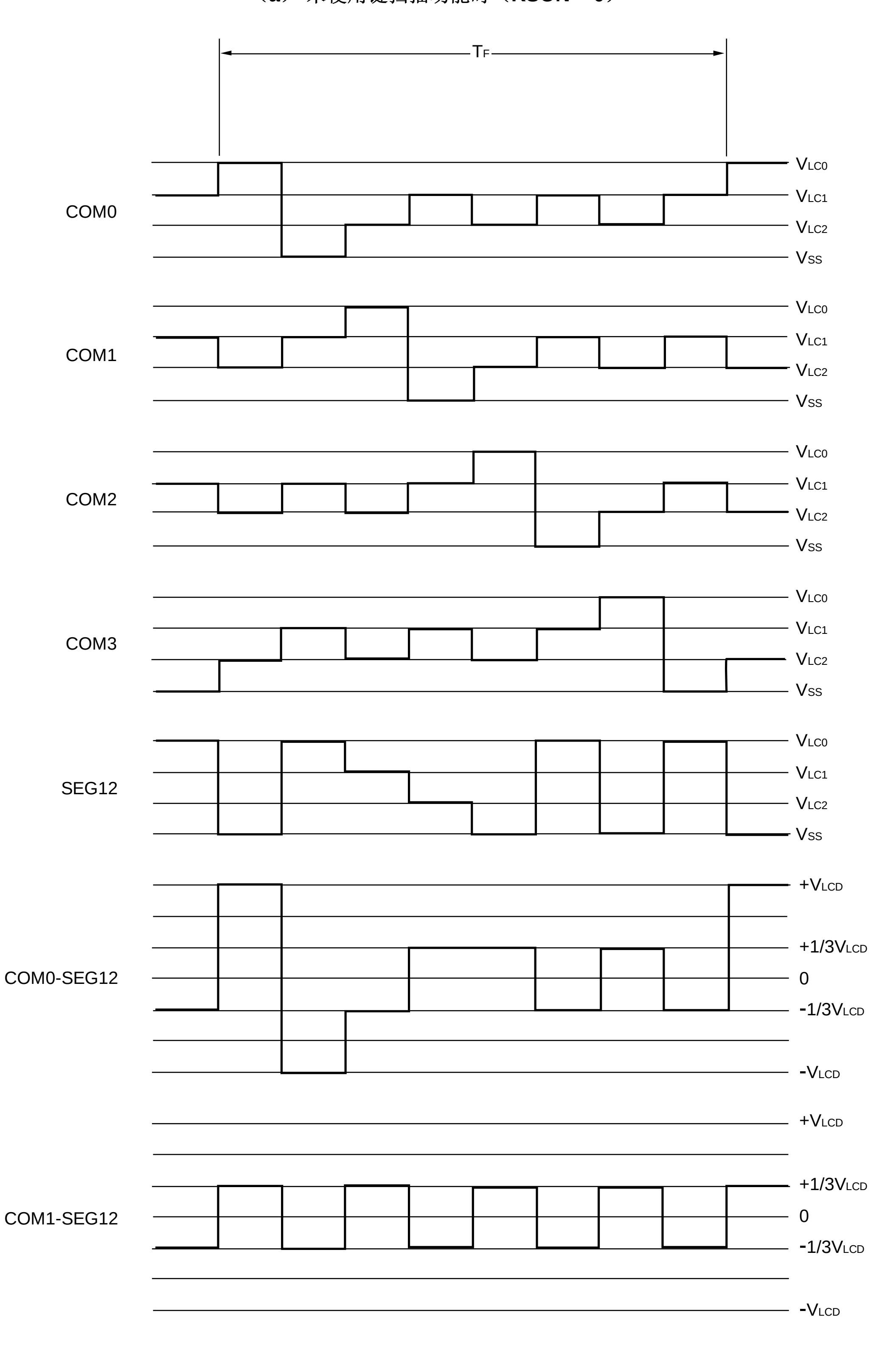


图 17-29. 4 分时 LCD 驱动波形示例（1/3 偏压方式）

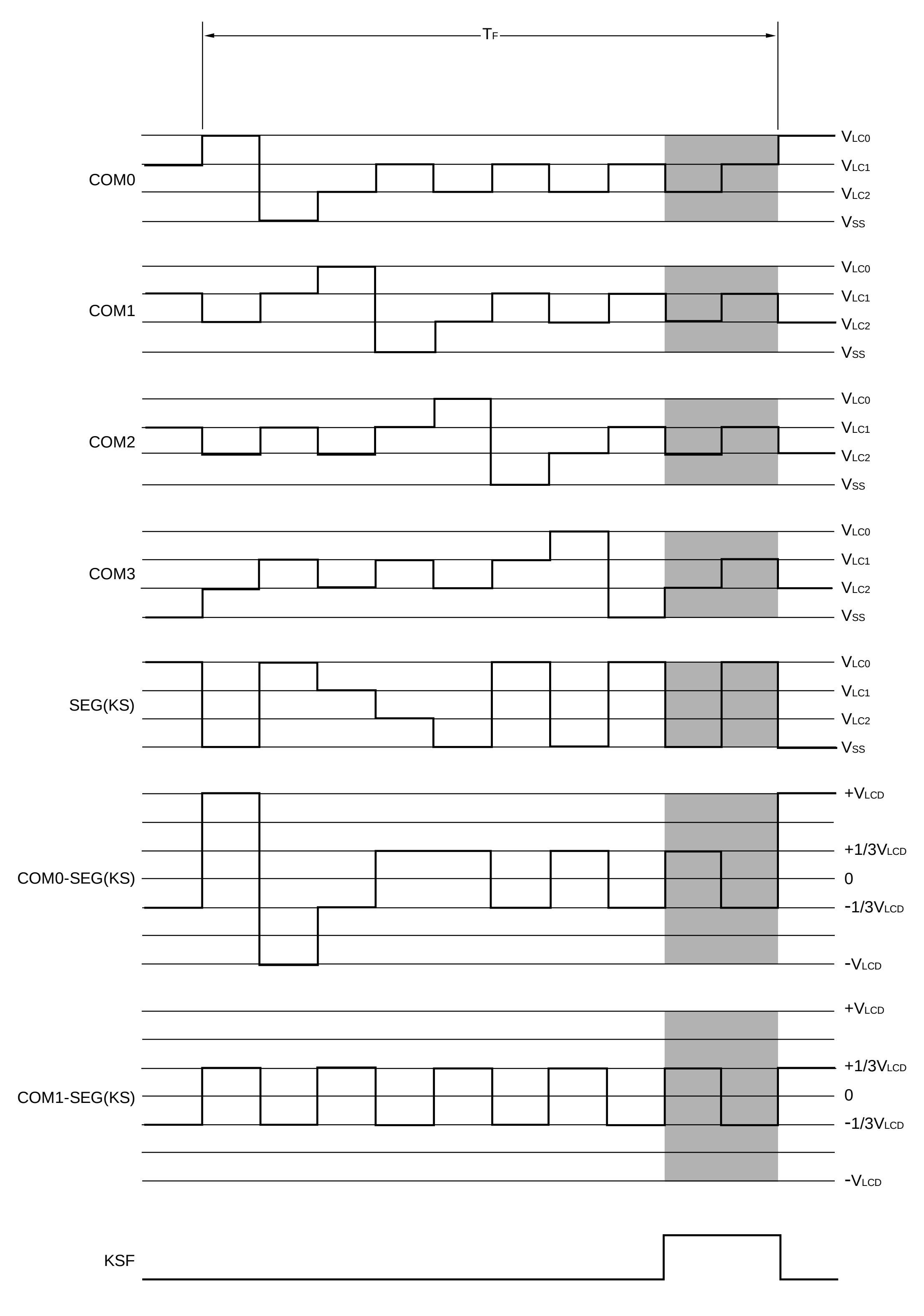
（a） 未使用键扫描功能时（KSON = 0）



备注 COM2 至 SEG12 和 COM3 至 SEG12 的波形被忽略。

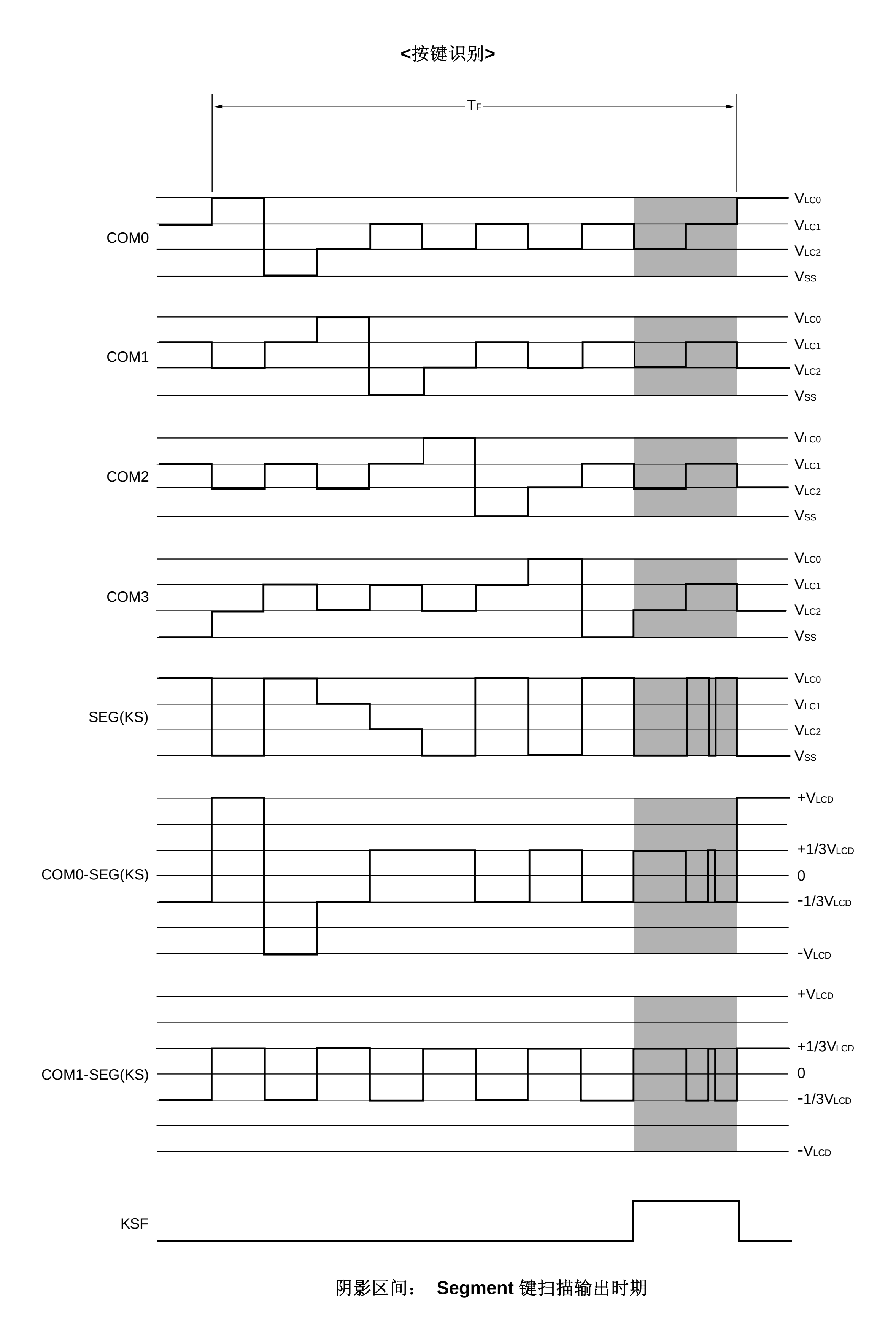
（**b**）　使用键扫描功能时（**KSON = 1**）

<按键输入等待>



阴影区间：　**Segment** 键扫描输出时期

<按键识别>



阴影区间： **Segment 键扫描输出时期**

备注 在按键识别期间，通过输出一个前半部和后半部具有反向关系的信号，可以消除 LCD 面板的剩余电荷。

17.7.5 8 分时显示示例

图 17-31 展示了如何把 15×8 点 LCD 面板连接到 78K0/LE3 芯片的 segment 信号（SEG4 至 SEG18）和公共端信号（COM0 至 COM7），该面板具有的显示模式如图 17-30 所示。这个示例在 LCD 面板上显示数字"123"。显示数据存储器（FA44H 至 FA52H 地址）的内容对应该显示。

以下描述重点关注显示在第一个字段的数字"3"（□）。为了在 LCD 面板显示"3"，必需在公共端信号 COM0 至 COM7 的时序，按照表 17-10 所示将选择电压或非选择电压输出到 SEG4 和 SEG8 引脚；segment 信号和 LCD 段的关系，参见图 **17-30**。

表 17-10. 选择和非选择电压（COM0 至 COM7）

Segment / 公共端	SEG4	SEG5	SEG6	SEG7	SEG8
COM0	选择	选择	选择	选择	选择
COM1	非选择	选择	非选择	非选择	非选择
COM2	非选择	非选择	选择	非选择	非选择
COM3	非选择	选择	非选择	非选择	非选择
COM4	选择	非选择	非选择	非选择	非选择
COM5	选择	非选择	非选择	非选择	选择
COM6	非选择	选择	选择	选择	非选择
COM7	非选择	非选择	非选择	非选择	非选择

根据表 17-10，它决定对应 SEG4 的显示数据存储器区域（FA44H）必须包含 00110001。

图 17-32 展示了 SEG4 和各个公共端信号之间的 LCD 驱动波形示例。在 COM0 的时序将选择电压输出到 SEG4 时，产生一个交替矩形波形，$+V_{LCD}/-V_{LCD}$，来打开对应的 LCD 段。

图 17-30. 8 分时 LCD 显示模型和电极连接

图 17-31. 8 分时 LCD 面板的连接示例

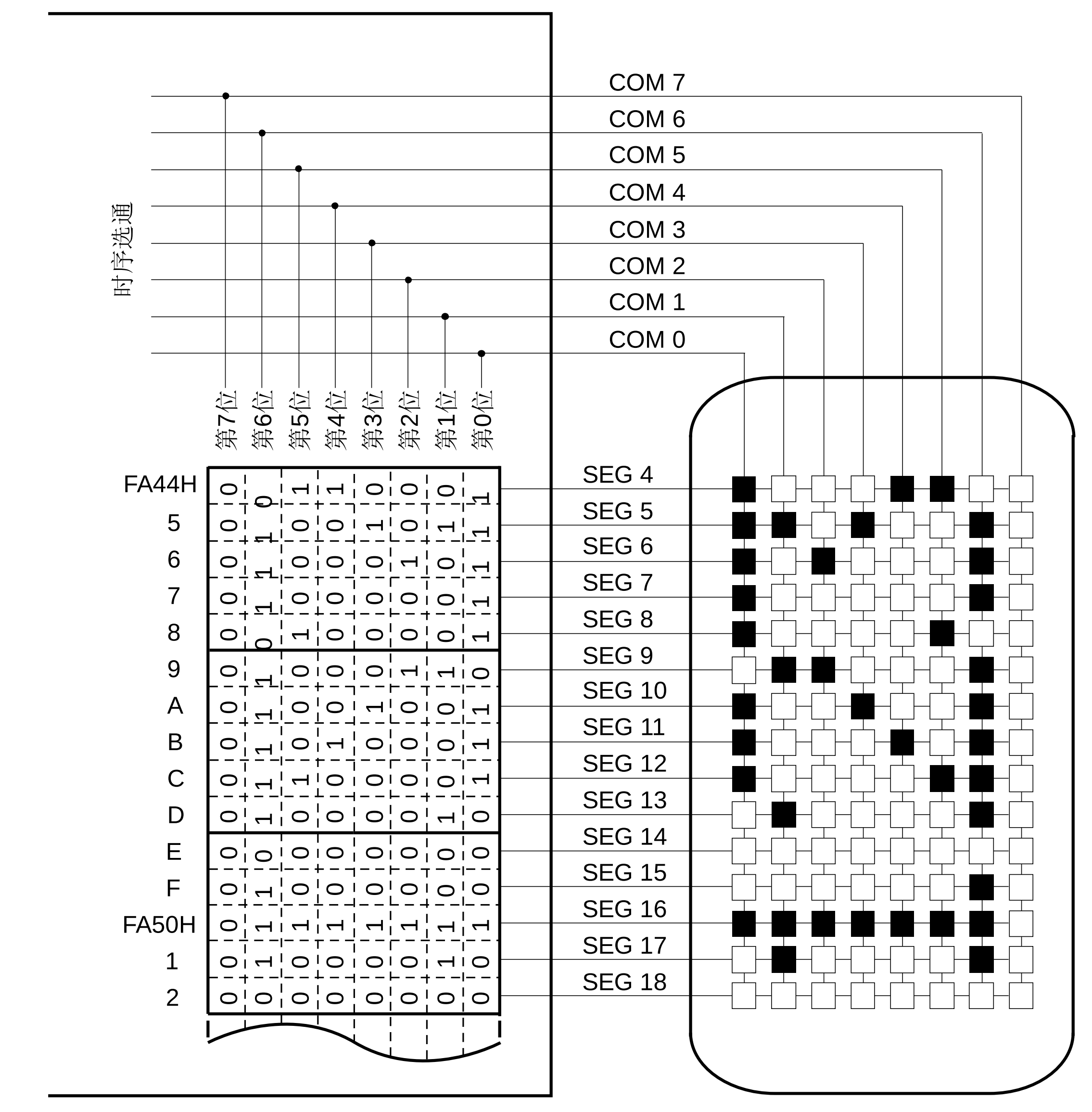
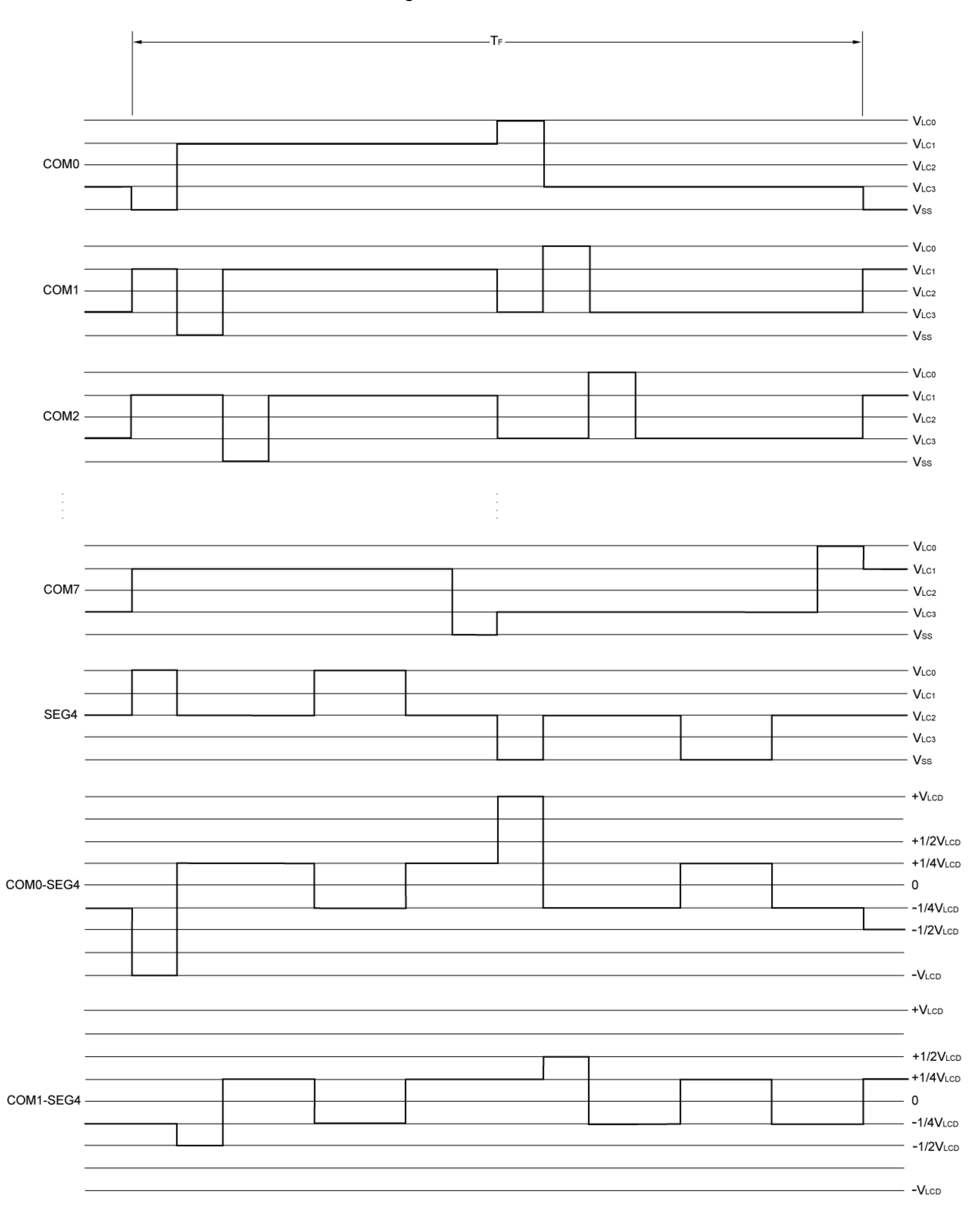


图 17-32. 8 分时 LCD 驱动波形示例（1/4 偏压方式）

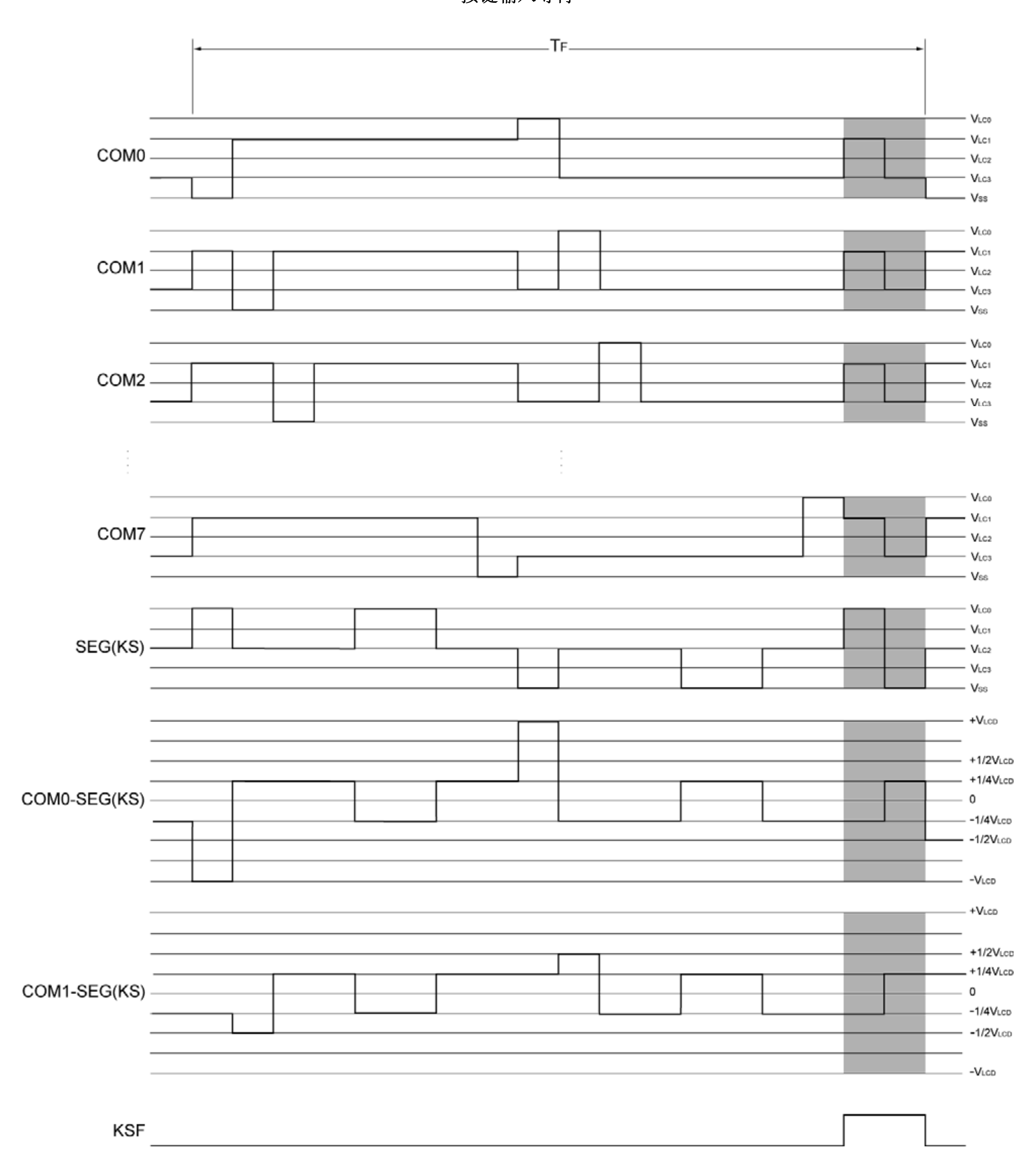
（a） 未使用 segment 键扫描功能时（KSON = 0）



备注　　COM3 至 COM6 的波形，COM2 至 SEG4 和 COM7 至 SEG4 被忽略。

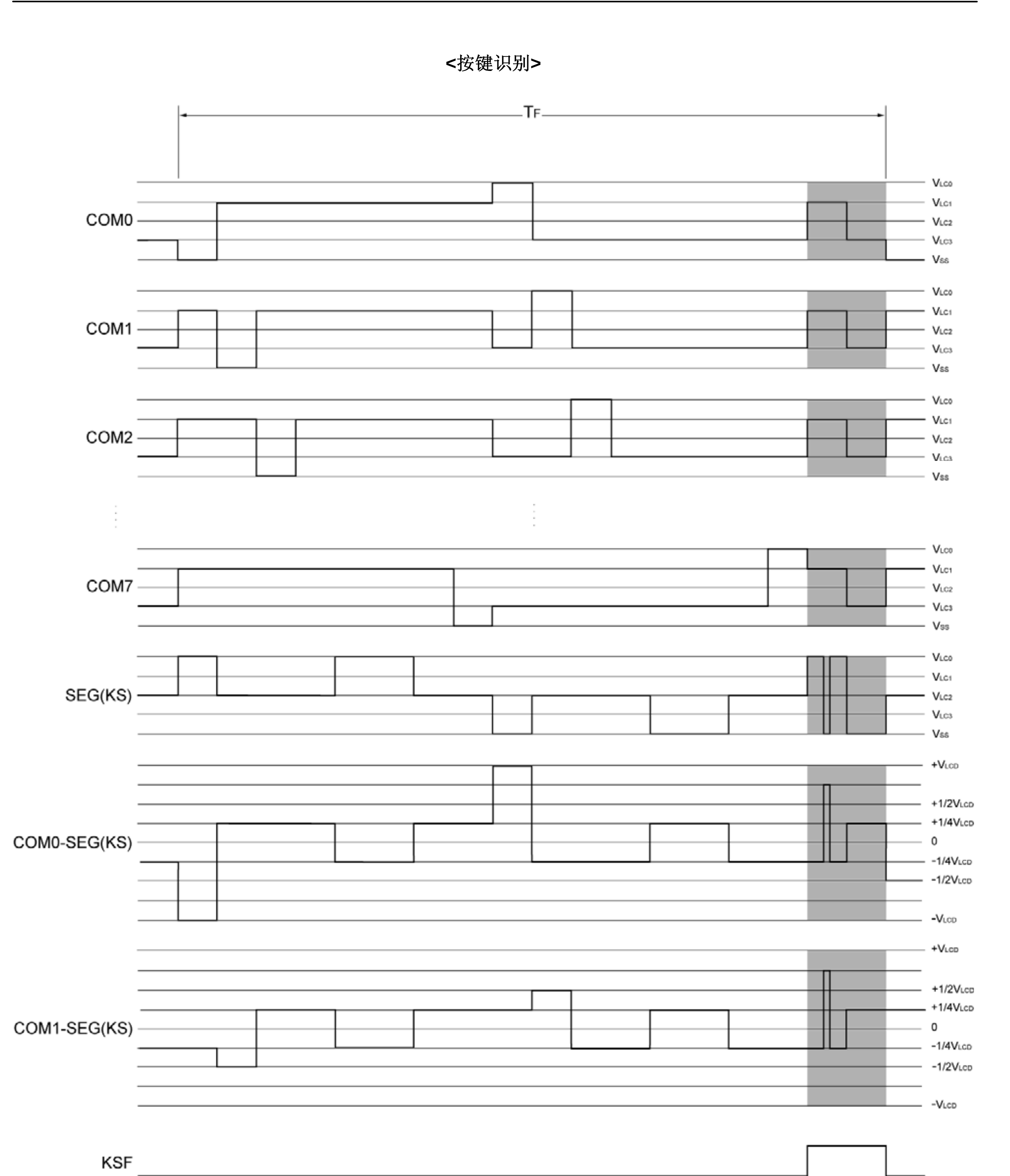
（**b**） 使用键扫描功能时（**KSON = 1**）

<按键输入等待>



阴影区间：　**Segment** 键扫描输出时期

<按键识别>



阴影区间：　**Segment** 键扫描输出时期

备注　　在按键识别期间，通过输出一个前半部和后半部具有反向关系的信号，可以消除 LCD 面板的剩余电荷。

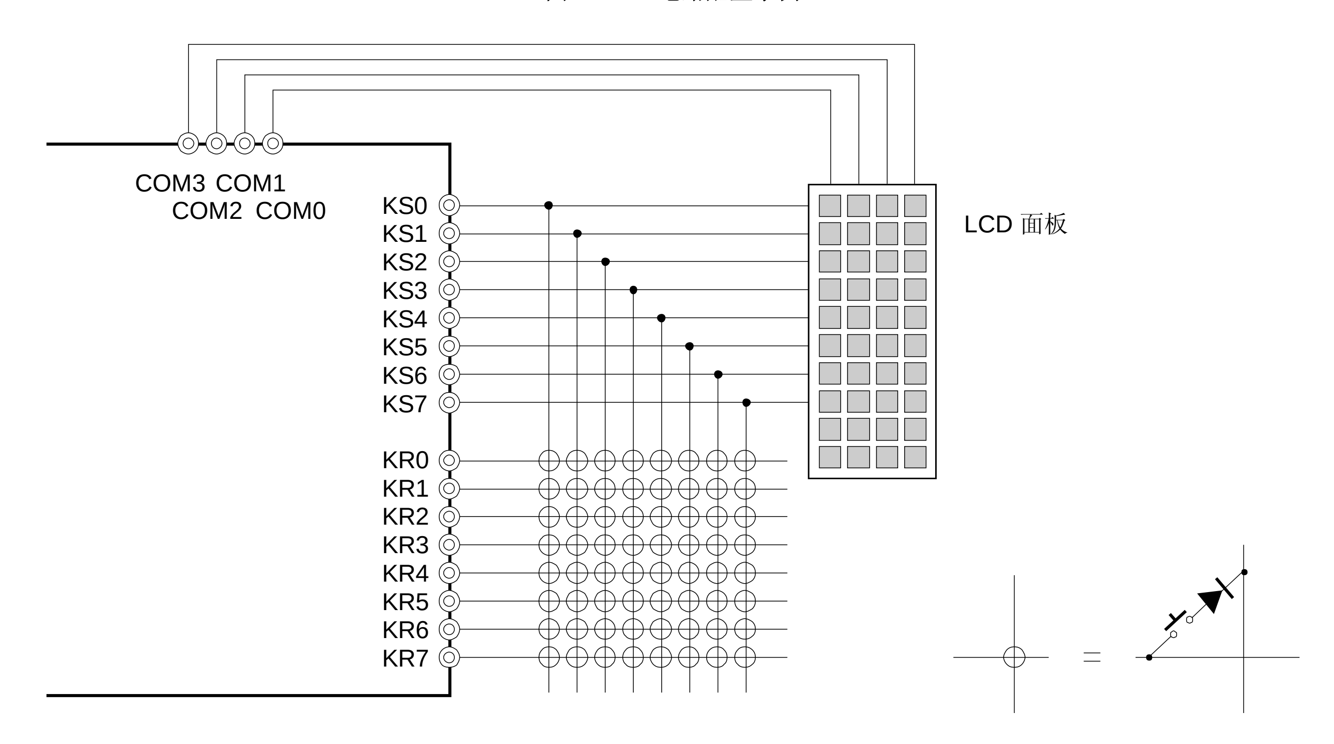
17.8 Segment 键扫描功能的操作

通过从相同的引脚输出 LCD 显示 segment 输出和键扫描信号，segment 键扫描功能可以减少使用引脚的数量。

注意事项 **该功能可能会影响 LCD 面板，取决于如何使用。**
请在全面评估后使用该功能。

17.8.1 电路配置示例

图 **17-33.** 电路配置示例



17.8.2 使用segment 键扫描功能的过程示例

图 17-34 展示了 segment 键扫描的操作流程，图 17-35 展示了按键连接示例。

图 17-34. Segment 键扫描的操作流程

开始

初始化设置

Segment 键扫描
(按键输入等待状态)

有键按下？
(KRIF = 1)

否

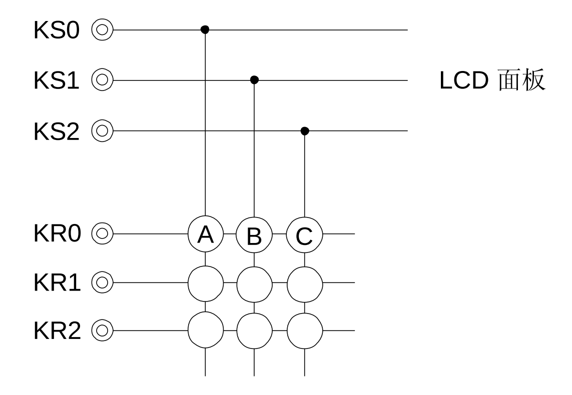
是

Segment 键扫描
(按键识别状态)

按键输入处理

INTKR 清除

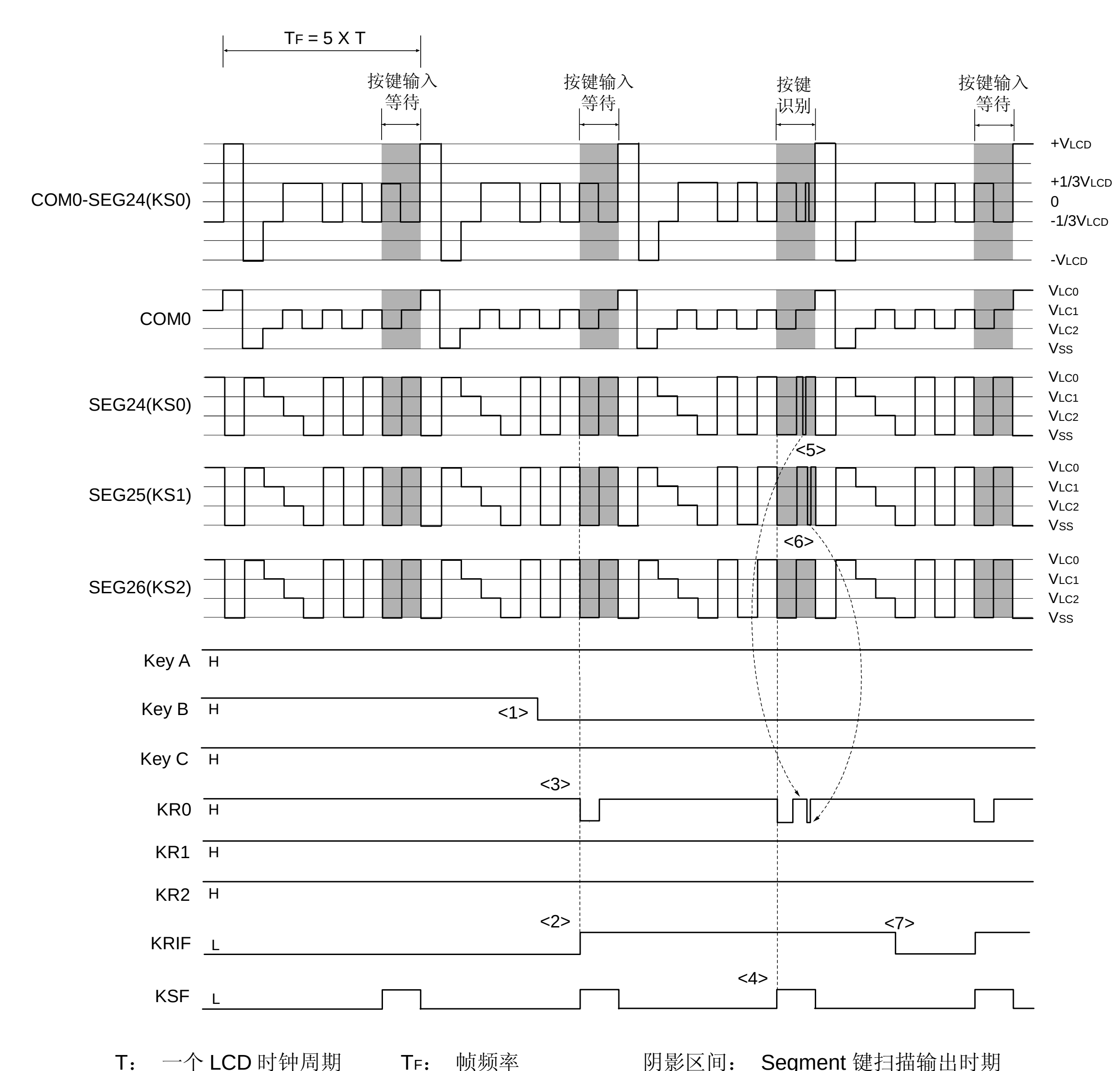
图 17-35. 按键连接示例



下面是当按键 B 被按下时 segment 键扫描操作的示例，如图 17-35 所示。

图 17-36. Segment 键扫描操作时序的示例

（4 分时（1/3 偏压方式））



T：　一个 LCD 时钟周期　　T_F：　帧频率　　阴影区间：　Segment 键扫描输出时期

<1>　假设按键 B 在这个时序被按下

<2>　KRIF 变为“1”，且可以知道有键被按下。

<3>　KR0 变为低电平，且可以知道按键 A，B 或 C 是否被按下过。
KSF 的上升沿之后，经过两个 f_{LCD} 时钟，输入到 KR 引脚将会被使能。于是，KSF 的上升沿之后经过两个 f_{LCD} 时钟，KRIF 变为“1”。

<4>　确认 KSF 为“1”之后，启动一个 segment 键扫描操作。

<5>　可以知道按键 A 没有被按下，因为当 SEG16（KS0）引脚输出低电平时，KR0 是高电平。

<6>　可以知道是按键 B 被按下，因为当 SEG17（KS1）引脚输出低电平时，KR0 是低电平。执行按键 B 的输入处理。

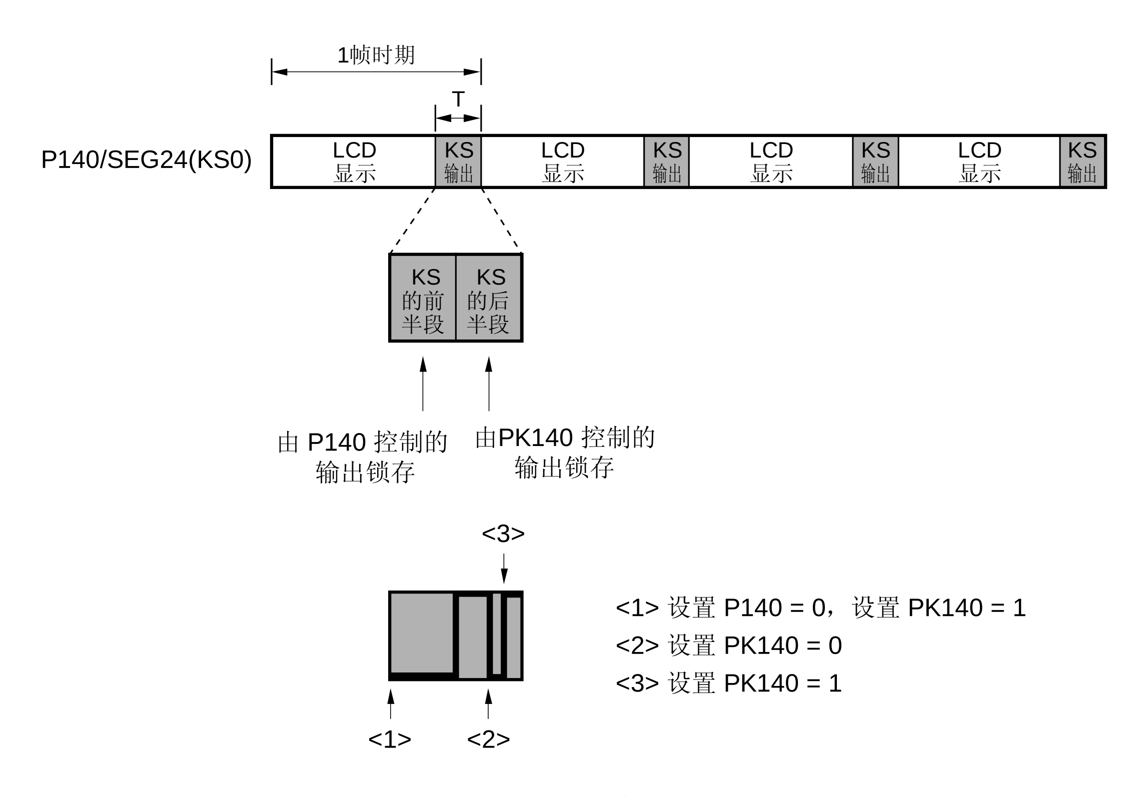
<7>　清除 KRIF。

在 segment 键扫描输出期间，SEG（KS）引脚的输出值对应端口寄存器 14 和 15 的设置值，并且可以通过端口寄存器 14 和 15 来控制。

各个端口寄存器的第 0 位至第 3 位和第 4 位至第 7 位分别被用来控制 segment 键扫描输出时期的前半部和后半部（参见 **17.3（9） 端口寄存器 14（P14）**和（**10） 端口寄存器 15（P15）**）。

图 17-37 展示了端口寄存器 14 和 segment 键扫描输出之间的关系。

图 17-37. 端口寄存器 14 和 Segment 键扫描输出之间的关系
（对于 P140/SEG16（KS0）引脚）



T : LCD 时钟的一个周期
LCD 显示 : LCD 指示信号输出时期
KS 输出 : Segment 键扫描输出时期

备注 segment 键扫描输出期间，输出时，COM 将不会被显示。
波形的详细情况，参见 图 **17-15** 和 **17-16**。

17.9 使用Segment 键扫描功能时的注意事项

（1） 使用条件

如果 V_{DD} 等于 V_{LC0}，使用 segment 键扫描功能。

（2） Segment 键扫描输入引脚

只有 KR0 至 KR4 引脚可以用作 segment 键扫描功能的输入引脚。

其他引脚不能用作 segment 键扫描功能的输入引脚

（3） KR0 至 KR4 引脚的允许输入范围

由于上拉电阻导致的延迟，从 segment 键扫描输出时期开始的两个 f_{LCD} 时钟时期内，不能在 KR 引脚执行 segment 键扫描输入。

与此类型，由于输入结束处理，segment 键扫描输出时期的最后一个 f_{LCD} 时钟内，不能在 KR 引脚执行 segment 键扫描。

（4） 按键返回模式寄存器（KRM） 设置

使用 segment 键扫描功能时（KSON = 1），设置 KRMn 为 1 或 0 来使用或不使用 KRn 引脚作为 segment 键扫描输入引脚。

（5） 电路配置

使用 segment 键扫描功能时，至少需要图 17-38 中所示的二极管 A 或二极管 B。

当二极管 A 或 B 缺失时，将会发生下列问题。

图 17-38. 按键矩阵配置示例

P140(KS0)
P141(KS1)
P152(KS6)
P153(KS7)
到 LCD 面板
二极管 A
P40/KR0
P41/KR1
到 KR
P46/KR6
P47/KR7
二极管 B
=

（a）当二极管 A 和 B 都缺失时

当二极管 A 和 B 都缺失时，因为如下原因，不能使用 segment 键扫描功能。

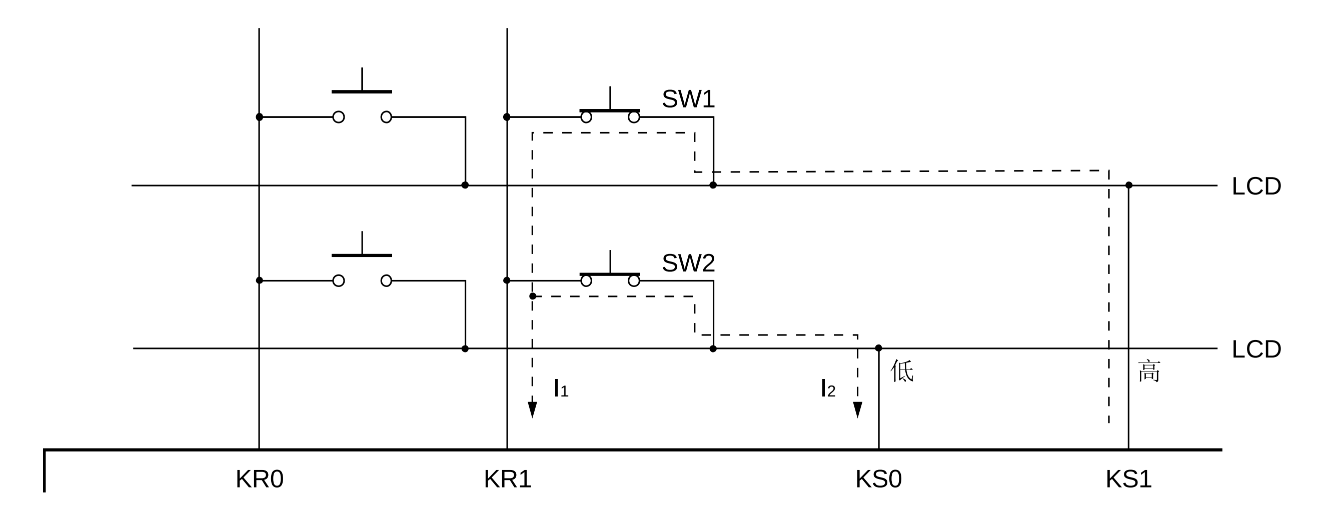
下图展示了当二极管 A 和 B 都缺失时的电路示例。

假设，如下图所示，开关 SW1 和 SW2 都被打开，分别从 KS1 引脚和 KS0 引脚输出高电平和低电平。

当此时缺失二极管 A 时，将会有电流 I_1 和 I_2，如虚线所示。

于是，因为 I_2，KS1 的高电平和 KS0 的低电平不会正常输出，并且 KR1 的按键输入数据将会变得不确定。

而且，LCD 显示不会被正常打开或关闭。



（b）当仅有二极管 A 存在时

当仅有二极管 A 存在时，因为如下原因，不能识别开关是否同时被按下。

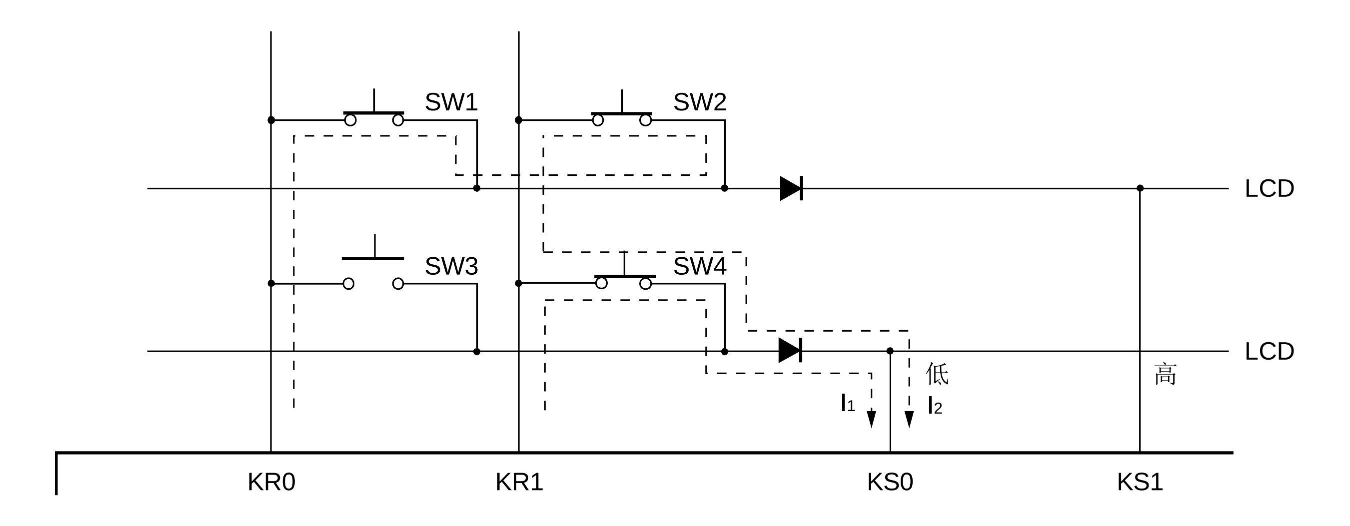
下图展示了当仅有二极管 A 存在时的电路示例。

假设，如下图所示，开关 SW1，SW2 和 SW4 被打开，分别从 KS1 引脚和 KS0 引脚输出高电平和低电平。

此时将会有电流 I_1 和 I_2，如虚线所示。

于是，即使 SW3 被关闭，SW3 也被识别为打开，因为 I_2，有低电平输入到 KR0。

对 LCD 显示没有干扰。



（c）当仅有二极管 B 存在时

不能执行三个以上开关被同时按下的识别。

对 LCD 显示没有干扰。

17.10 LCD驱动电压V_{LC0}，V_{LC1}，V_{LC2}和V_{LC3}

78K0/LE3 中，LCD 驱动电源提供可由 2 种方式产生：内部电阻分压方式或外部电阻分压方式。

17.10.1 内部电阻分压方式

78K0/LE3 内置有分压电阻器可以产生 LCD 驱动电源。使用内部分压电阻器，可以产生的 LCD 驱动电源能够满足表 17-11 列举的各种偏压模式，而无需使用外部分压电阻器。

表 17-11. LCD 驱动电压（具有片上分压电阻）

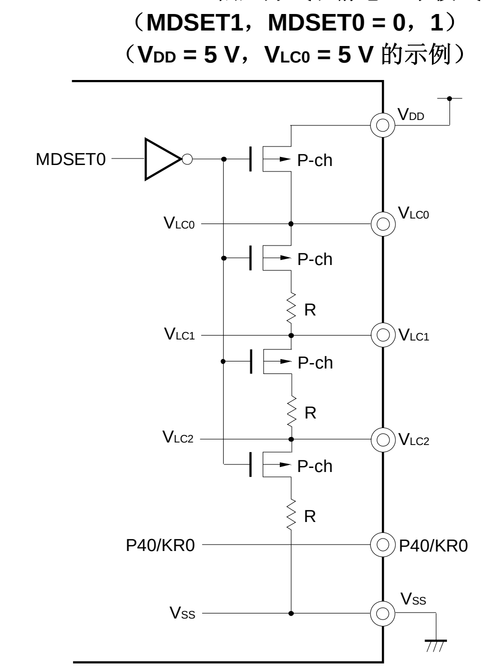
偏压方式 / LCD 驱动电压引脚	无偏压（静态）	1/2 偏压方式	1/3 偏压方式	1/4 偏压方式
V_{LC0}	V_{LCD}	V_{LCD}	V_{LCD}	V_{LCD}
V_{LC1}	$\frac{2}{3}V_{LCD}$	$\frac{1}{2}V_{LCD}$ 注	$\frac{2}{3}V_{LCD}$	$\frac{3}{4}V_{LCD}$
V_{LC2}	$\frac{1}{3}V_{LCD}$		$\frac{1}{3}V_{LCD}$	$\frac{2}{4}V_{LCD}$
V_{LC3}	V_{SS}	V_{SS}	V_{SS}	$\frac{1}{4}V_{LCD}$

注 对于 1/2 偏压模式，必需外部连接 V_{LC1} 和 V_{LC2} 引脚。

图 17-39 展示了根据表 17-11 内部产生 LCD 驱动电压的示例。

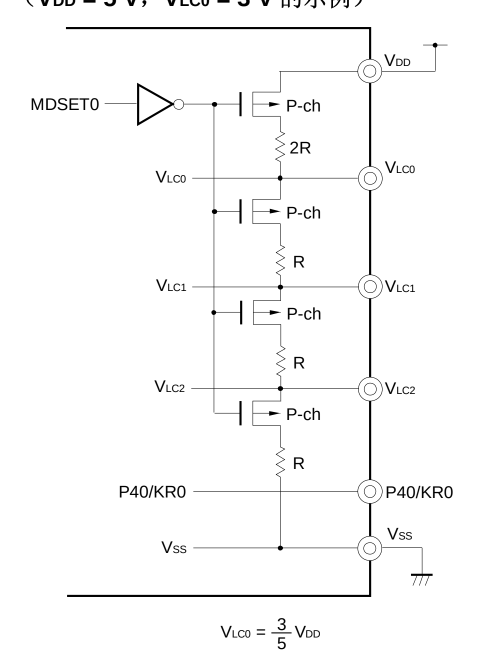
图 17-39. LCD 驱动电源连接的示例（内部电阻分压方式）（1/2）

（a） 1/3 偏压方式和静态显示模式
（MDSET1，MDSET0 = 0，1）
（V_{DD} = 5 V，V_{LC0} = 5 V 的示例）



$V_{LC0} = V_{DD}$

（b） 1/3 偏压方式和静态显示模式
（MDSET1，MDSET0 = 1，1）
（V_{DD} = 5 V，V_{LC0} = 3 V 的示例）



$V_{LC0} = \frac{3}{5}V_{DD}$

备注 当使用静态显示模式时，为了通过分压电阻减少功耗，推荐使用外部分压电阻方式。

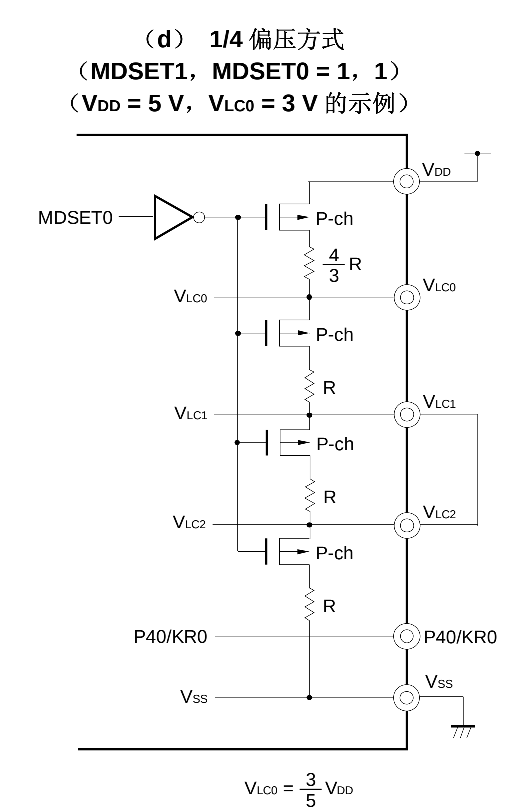
图 17-39. LCD 驱动电源连接的示例（内部电阻分压模式）（2/2）

（c） **1/2** 偏压方式
（**MDSET1，MDSET0 = 0，1**）
（V_{DD} **= 5 V，**V_{LC0} **= 5 V** 的示例）

MDSET0
V_{DD}
P-ch
V_{LC0}
P-ch
R
V_{LC1}
P-ch
R
V_{LC2}
P-ch
R
P40/KR0
V_{SS}

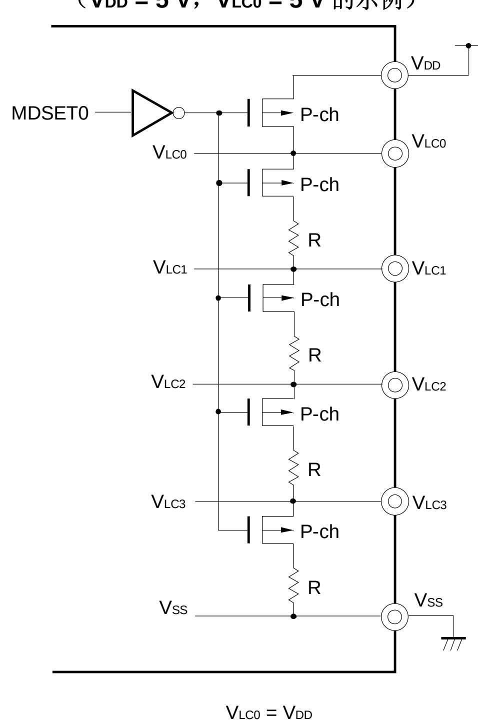
$V_{LC0} = V_{DD}$

（d） **1/4** 偏压方式
（**MDSET1，MDSET0 = 1，1**）
（V_{DD} **= 5 V，**V_{LC0} **= 3 V** 的示例）



$V_{LC0} = \frac{3}{5} V_{DD}$

（e） **1/4** 偏压方式
（**MDSET1，MDSET0 = 0，1**）
（V_{DD} **= 5 V，**V_{LC0} **= 5 V** 的示例）



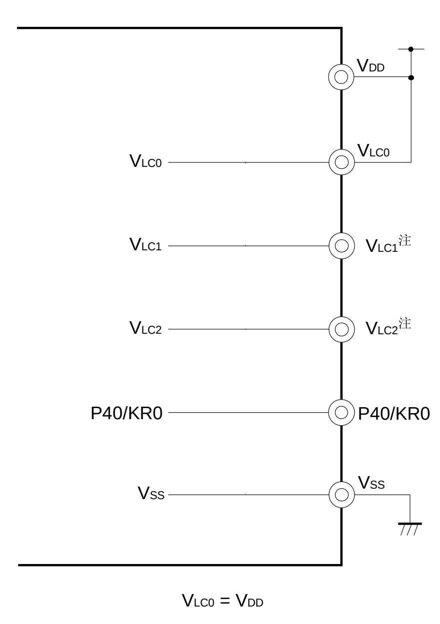
$V_{LC0} = V_{DD}$

17.10.2 外部电阻分压方式

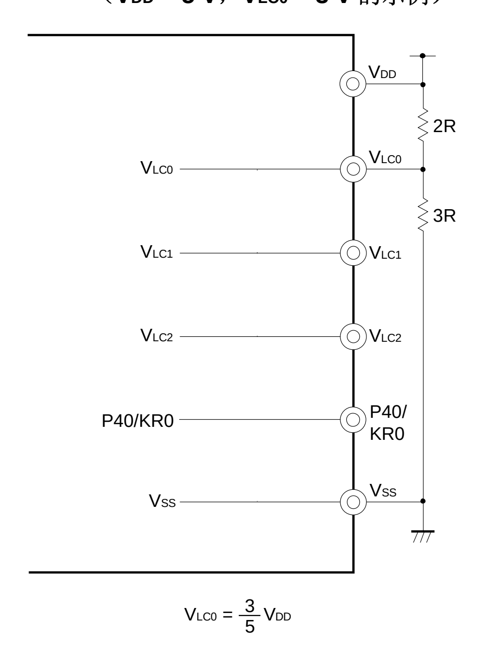
78K0/LE3 也可以使用外部分压电阻产生 LCD 驱动电源，而不使用内部电阻。图 17-40 展示了对应于各种偏压方式的 LCD 驱动电压连接示例。

图 17-40. LCD 驱动电压连接示例（外部电阻分压模式）（1/2）

（a）静态显示模式
（MDSET1，MDSET0 = 0，0）
（V_{DD} = 5 V，V_{LC0} = 5 V 的示例）

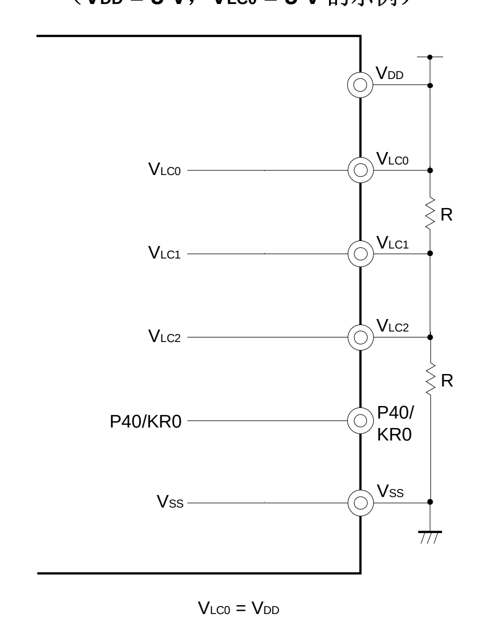


（b）静态显示模式
（MDSET1，MDSET0 = 0，0）
（V_{DD} = 5 V，V_{LC0} = 3 V 的示例）



注　直接将 V_{LC1} 和 V_{LC2} 连接到 GND 或 V_{LC0}。

（c）1/2 偏压方式
（MDSET1，MDSET0 = 0，0）
（V_{DD} = 5 V，V_{LC0} = 5 V 的示例）



（d）1/2 偏压方式
（MDSET1，MDSET0 = 0，0）
（V_{DD} = 5 V，V_{LC0} = 3 V 的示例）

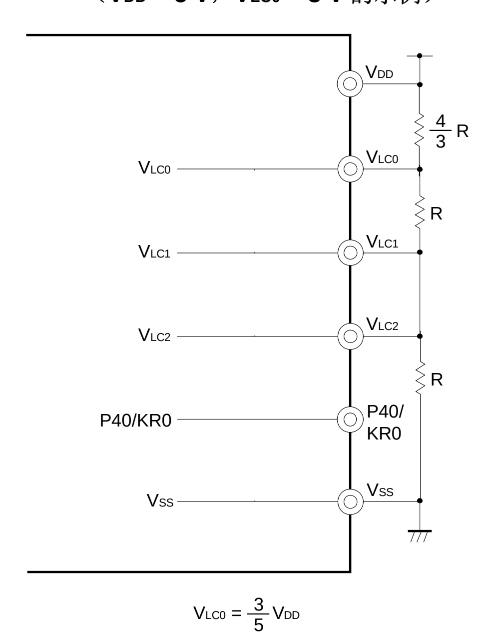
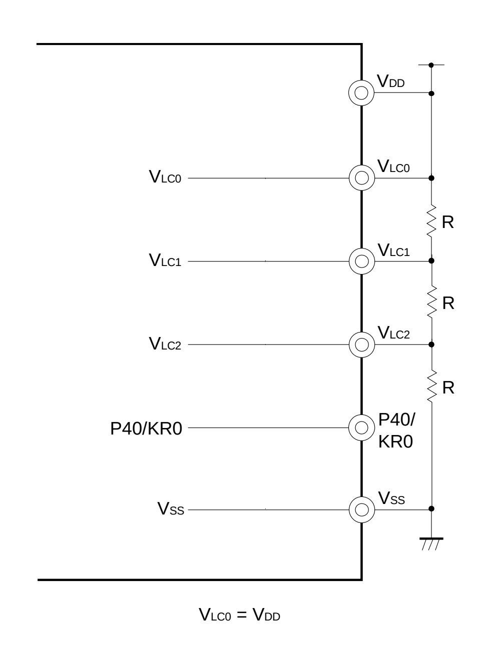
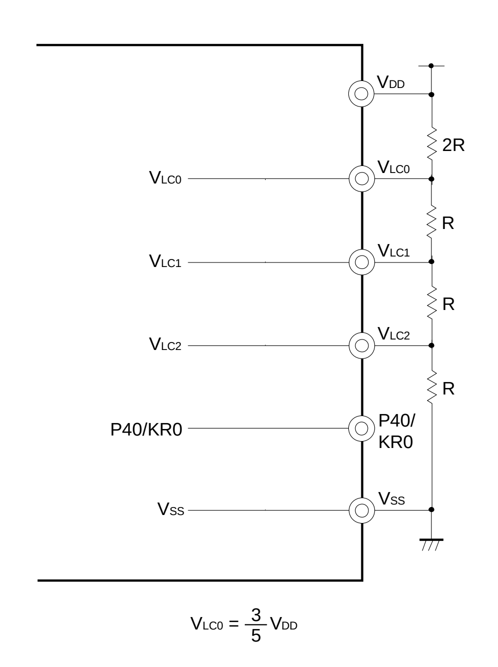


图 17-40. LCD 驱动电压连接示例（外部电阻分压模式）（2/2）

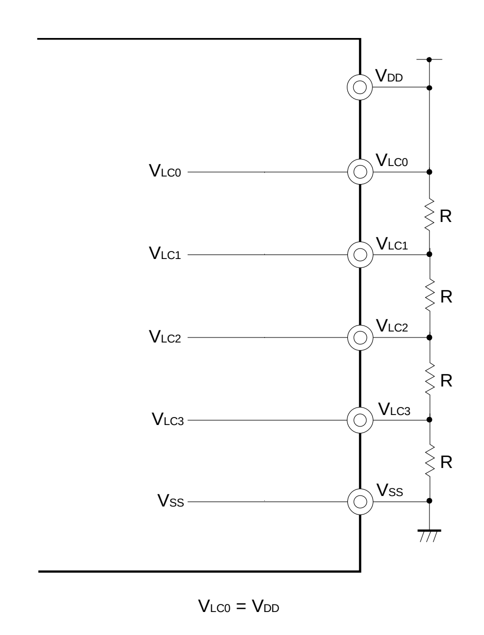
（e） 1/3 偏压方式
（MDSET1，MDSET0 = 0，0）
（V_{DD} = 5 V，V_{LC0} = 5 V 的示例）



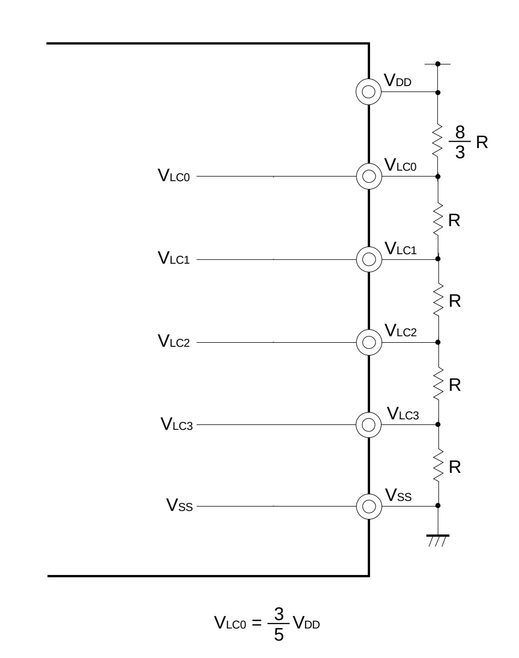
（f） 1/3 偏压方式
（MDSET1，MDSET0 = 0，0）
（V_{DD} = 5 V，V_{LC0} = 3 V 的示例）



（g） 1/4 偏压方式
（MDSET1，MDSET0 = 0，0）
（V_{DD} = 5 V，V_{LC0} = 5 V 的示例）



（h） 1/4 偏压方式
（MDSET1，MDSET0 = 0，0）
（V_{DD} = 5 V，V_{LC0} = 3 V 的示例）



18.1 曼彻斯特编码发生器的功能

曼彻斯特编码发生器具有以下三种模式可供使用。

（1）　操作停止模式

不执行曼彻斯特编码发生器/位序列缓冲器的输出时，使用该模式。该模式可以降低功耗。

详情参见 **18.4.1 操作停止模式**。

（2）　曼彻斯特编码发生器模式

该模式用于从 MCGO 引脚发送曼彻斯特码。

发送的位长度可以设置，并且使能可变位长度的传输。同样，对于 8 位传输数据，可以设置数据传输的输出电平和 LSB-先行或 MSB-先行。

（3）　位序列缓冲器模式

该模式用于从 MCGO 引脚发送位序列数据。

发送的位长度可以设置，并且使能可变位长度的传输。同样，对于 8 位传输数据，可以设置数据传输的输出电平和 LSB-先行或 MSB-先行。

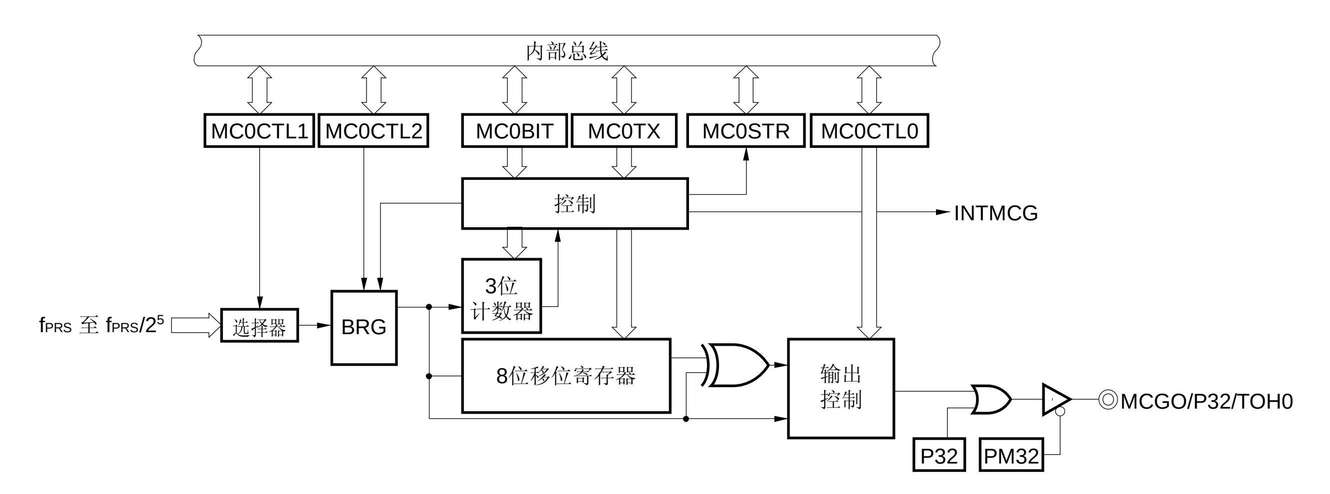
18.2 曼彻斯特编码发生器的配置

曼彻斯特编码发生器包括如下硬件。

表 18-1. 曼彻斯特编码发生器的配置

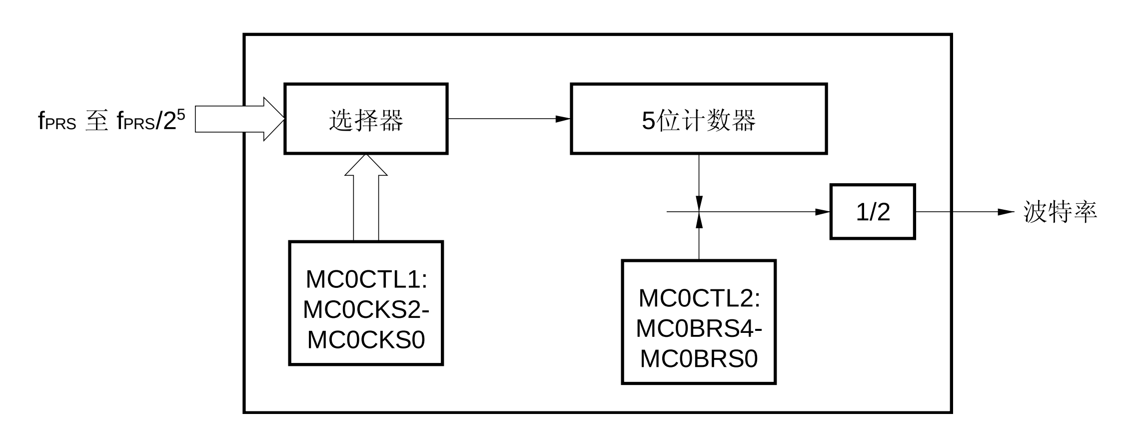
项目	配置
寄存器	MCG 发送缓冲器寄存器（MC0TX） MCG 发送位计数规范寄存器（MC0BIT）
控制寄存器	MCG 控制寄存器 0 （MC0CTL0） MCG 控制寄存器 1 （MC0CTL1） MCG 控制寄存器 2 （MC0CTL2） MCG 状态寄存器（MC0STR） 端口模式寄存器 3 （PM3） 端口寄存器 3 （P3）

图 18-1. 曼彻斯特编码发生器的框图



备注		
	BRG：	波特率发生器
	f_{PRS}：	外设硬件时钟频率
	MC0BIT：	MCG 发送位计数规范寄存器
	MC0CTL2 至 MC0CTL0：	MCG 控制寄存器 2 至 0
	MC0STR：	MCG 状态寄存器
	MC0TX：	MCG 发送缓冲器寄存器

图 18-2. 波特率发生器的框图



备注		
	f_{PRS}：	外设硬件时钟频率
	MC0CTL2，MC0CTL1：	MCG 控制寄存器 2，1
	MC0CKS2 至 MC0CKS0：	MC0CTL1 寄存器的第 2 位至第 0 位
	MC0BRS4 至 MC0BRS0：	MC0CTL2 寄存器的第 4 位至第 0 位

（1）　MCG 发送缓冲器寄存器（MC0TX）

该寄存器用于设置发送数据。MCG 控制寄存器 0 （MC0CTL0）的第 7 位（MC0PWR）为 1，当数据写入 MC0TX 时启动发送操作。

写入 MC0TX 的数据通过 8 位移位寄存器转换为串行数据，并且输出到 MCGO 引脚。

使用 MCG 控制寄存器 0 （MC0CTL0）的第 1 位，可以将曼彻斯特编码或位序列数据设置为输出码。

可以通过 8 位存储器操作指令来设置该寄存器。

复位信号的发生会将该寄存器设置为 FFH。

（2）　MCG 发送位计数规范寄存器（MC0BIT）

该寄存器用于设置发送位的数量。

设置发送数据到 MC0TX 之前，设置发送位计数到该寄存器。

连续发送时，在产生发送开始中断（INTMCG）后，需要将下一个待发送的发送位数量写入。但是，如果下一个发送计数的数量与上一个发送计数相同，则该寄存器不需要写入。

可以通过 8 位存储器操作指令来设置该寄存器。

复位信号的发生会将该寄存器设置为 07H。

图 18-3. MCG 发送位计数规范寄存器（MC0BIT）的格式

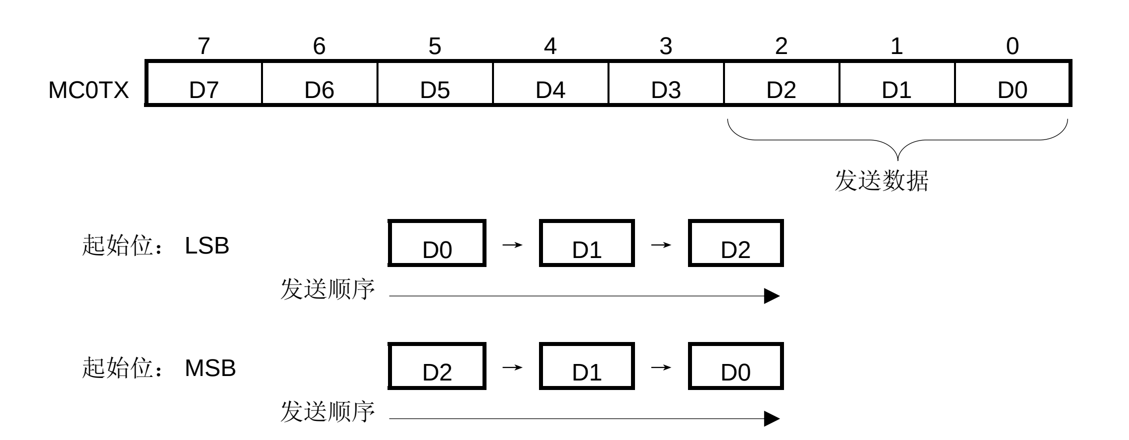
地址：　FF4BH　复位后：　07H　R/W

符号	7	6	5	4	3	<2>	<1>	<0>
MC0BIT	0	0	0	0	0	MC0BIT2	MC0BIT1	MC0BIT0

MC0BIT2	MC0BIT1	MC0BIT0	发送位计数设置
0	0	0	1 位
0	0	1	2 位
0	1	0	3 位
0	1	1	4 位
1	0	0	5 位
1	0	1	6 位
1	1	0	7 位
1	1	1	8 位

备注　当发送位的数量被设置为 7 位或更少时，无论设置 MSB/LSB 作为起始位，低位总是被发送。

例。　当发送位的数量被设置为 3 位，并且 D7 至 D0 被写入到 MCG 发送缓冲寄存器（MC0TX）



18.3 控制曼彻斯特编码发生器的寄存器

下面六种类型的寄存器用于控制曼彻斯特编码发生器。

- MCG 控制寄存器 0 （MC0CTL0）
- MCG 控制寄存器 1 （MC0CTL1）
- MCG 控制寄存器 2 （MC0CTL2）
- MCG 状态寄存器 （MC0STR）
- 端口模式寄存器 3 （PM3）
- 端口寄存器 3 （P3）

（1）　MCG 控制寄存器 0 （MC0CTL0）

该寄存器用于设置操作模式和操作的使能/禁止。

可以通过 1 位或 8 位存储器操作指令来设置该寄存器。

复位信号的发生会将该寄存器设置为 10H。

图 18-4. MCG 控制寄存器 0（MC0CTL0）的格式

地址： FF4CH　复位后： 10H　R/W

符号	<7>	6	5	<4>	3	2	<1>	<0>
MC0CTL0	MC0PWR	0	0	MC0DIR	0	0	MC0OSL	MC0OLV

MC0PWR	操作控制
0	操作停止
1	操作使能

MC0DIR	首位规范
0	MSB
1	LSB

MC0OSL	数据格式
0	曼彻斯特编码
1	位序列数据

MC0OLV	当发送暂停时的输出电平
0	低电平
1	高电平

注意事项　**在重写 MC0DIR，MC0OSL 和 MC0OLV 位（当 MC0PWR 位被置位（1）时的同时，可以通过 8 位存储器操作指令重写这些位）之前，清除（0）MC0PWR 位。**

（2）　MCG 控制寄存器 1（MC0CTL1）

该寄存器用于设置曼彻斯特编码发生器的基准时钟。

可以通过 8 位存储器操作指令来设置该寄存器。

复位信号的发生会将该寄存器清除为 00H。

图 18-5. MCG 控制寄存器 1（MC0CTL1）的格式

地址： FF4DH　复位后： 00H　R/W

符号	7	6	5	4	3	2	1	0
MC0CTL1	0	0	0	0	0	MC0CKS2	MC0CKS1	MC0CKS0

MC0CKS2	MC0CKS1	MC0CKS0	基准时钟（f_{XCLK}）的选择[注 1]
0	0	0	f_{PRS}[注 2]（10 MHz）
0	0	1	$f_{PRS}/2$ （5 MHz）
0	1	0	$f_{PRS}/2^2$ （2.5 MHz）
0	1	1	$f_{PRS}/2^3$ （1.25 MHz）
1	0	0	$f_{PRS}/2^4$ （625 kHz）
1	0	1	$f_{PRS}/2^5$ （312.5 kHz）
1	1	0	禁止设置
1	1	1	

注　**1.** 如果外设硬件时钟（f_{PRS}）运行于高速系统时钟（f_{XH}）（XSEL ＝ 1），根据供电电压的不同，f_{PRS} 的工作频率也不同。

- V_{DD} = 2.7 至 5.5 V：　$f_{PRS} \leq 10$ MHz
- V_{DD} = 1.8 至 2.7 V：　$f_{PRS} \leq 5$ MHz

2. 如果外部硬件时钟 （f_{PRS}）运行于内部高速振荡时钟（f_{RH}）（XSEL = 0），当 1.8 V ≤ V_{DD} < 2.7 V 时，禁止设定 MC0CKS2 = MC0CKS1 = MC0CKS0 = 0（基准时钟： f_{PRS}）。

注意事项　在重写 **MC0CKS2** 至 **MC0CKS0** 位之前，将 **MC0CTL0** 寄存器的第 **7** 位（**MC0PWR**）清除为 **0**。

备注　**1.** f_{PRS}：　外设硬件时钟频率

2. 括号内的数字适用于 f_{PRS} = 10 MHz。

（3）　MCG 控制寄存器 2 （MC0CTL2）

该寄存器用于设置发送波特率。

可以通过 8 位存储器操作指令来设置该寄存器。

复位信号的发生会将该寄存器设置为 1FH。

图 18-6. MCG 控制寄存器 2（MC0CTL2）的格式

地址： FF4EH　复位后： 1FH　R/W

符号	7	6	5	4	3	2	1	0
MC0CTL2	0	0	0	MC0BRS4	MC0BRS3	MC0BRS2	MC0BRS1	MC0BRS0

MC0BRS4	MC0BRS3	MC0BRS2	MC0BRS1	MC0BRS0	k	5 位计数器的输出时钟选择
0	0	0	×	×	4	$f_{XCLK}/4$
0	0	1	0	0	4	$f_{XCLK}/4$
0	0	1	0	1	5	$f_{XCLK}/5$
0	0	1	1	0	6	$f_{XCLK}/6$
0	0	1	1	1	7	$f_{XCLK}/7$
• • • • •	• • • • •	• • • • •	• • • • •	• • • • •	• • • • •	• • • • •
1	1	1	0	0	28	$f_{XCLK}/28$
1	1	1	0	1	29	$f_{XCLK}/29$
1	1	1	1	0	30	$f_{XCLK}/30$
1	1	1	1	1	31	$f_{XCLK}/31$

注意事项　1.　在重写 MC0BRS4 至 MC0BRS0 位之前，将 MC0CTL0 寄存器的第 7 位（MC0PWR）清除为 0。

2.　5 位计数器的输出时钟除 2 得到的值是波特率值。

备注　**1.** f_{XCLK}：　由 MC0CTL1 寄存器的 MC0CKS2 至 MC0CKS0 位选择的基准时钟的频率。

2. k：　由 MC0BRS4 至 MC0BRS0 位设置的值（k = 4，5，6，7，….，31）

3. ×：　无须理会

（4）　MCG 状态寄存器 （MC0STR）

该寄存器用于表示曼彻斯特编码发生器的操作状态。

可以通过 1 位或 8 位存储器操作指令来设置该寄存器。不能写入该寄存器。

复位信号的发生或设置 MC0PWR = 0，会将该寄存器清除为 00H。

图 **18-7. MCG** 状态寄存器（**MC0STR**）的格式

地址： FF47H　复位后： 00H　R

符号	<7>	6	5	4	3	2	1	0
MC0STR	MC0TSF	0	0	0	0	0	0	0

MC0TSF	数据发送状态
0	• 复位信号产生 • MC0PWR = 0 • 当一个发送完成时，如果下一个发送数据没有写入到 MC0TX
1	发送操作进行中

注意事项　在连续发送期间，该标志总是表示 1。没有确定此标志被清除，则不要初始化发送操作

18.4 曼彻斯特编码发生器的操作

曼彻斯特编码发生器有下面描述的三种模式。

- 操作停止模式
- 曼彻斯特编码发生器模式
- 位序列缓冲器模式

18.4.1 操作停止模式

在操作停止模式下，不执行发送。因此，功耗可以降低。此外，P32/TOH0/MCGO 引脚在该模式下被用作普通 I/O 端口。

（1）　寄存器描述

MCG 控制寄存器 0（MC0CTL0）用于设置操作停止模式。

要设置操作停止模式，将 MC0CTL0 的第 7 位（MC0PWR）清除为 0。

（a） MCG 控制寄存器 0（MC0CTL0）

可以通过 1 位或 8 位存储器操作指令来设置该寄存器。

复位信号的发生会将该寄存器设置为 10H。

地址： FF4CH　复位后： 10H　R/W

符号	<7>	6	5	<4>	3	2	<1>	<0>
MC0CTL0	MC0PWR	0	0	MC0DIR	0	0	MC0OSL	MC0OLV

MC0PWR	操作控制
0	操作停止

18.4.2 曼彻斯特编码发生器模式

该模式用于使用 MCGO 引脚按照曼彻斯特编码格式发送数据。

（1）　寄存器描述

MCG 控制寄存器 0（MC0CTL0），MCG 控制寄存器 1（MC0CTL1），和 MCG 控制寄存器 2（MC0CTL2）用于设置曼彻斯特编码发生器模式。

（a） MCG 控制寄存器 0（MC0CTL0）

可以通过 1 位或 8 位存储器操作指令来设置该寄存器。

复位信号的发生会将该寄存器设置为 10H。

地址： FF4CH　复位后： 10H　R/W

符号	<7>	6	5	<4>	3	2	<1>	<0>
MC0CTL0	MC0PWR	0	0	MC0DIR	0	0	MC0OSL	MC0OLV

MC0PWR	操作控制
0	操作停止
1	操作允许

MC0DIR	首位规范
0	MSB
1	LSB

MC0OSL	数据格式
0	曼彻斯特编码
1	位序列数据

MC0OLV	当发送暂停时的输出电平
0	低电平
1	高电平

注意事项　在重写 MC0DIR，MC0OSL 和 MC0OLV 位（当 MC0PWR 位被置位（1）时的同时，可以通过 8 位存储器操作指令重写这些位）之前，清除（0）MC0PWR 位。

（b） MCG 控制寄存器 1 （MC0CTL1）

该寄存器用于设置曼彻斯特编码发生器的基准时钟。

可以通过 8 位存储器操作指令来设置该寄存器。

复位信号的发生会将该寄存器清除为 00H。

地址： FF4DH 复位后： 00H R/W

符号	7	6	5	4	3	2	1	0
MC0CTL1	0	0	0	0	0	MC0CKS2	MC0CKS1	MC0CKS0

MC0CKS2	MC0CKS1	MC0CKS0	基准时钟（f_{XCLK}）的选择[注 1]
0	0	0	f_{PRS}[注 2] （10 MHz）
0	0	1	$f_{PRS}/2$ （5 MHz）
0	1	0	$f_{PRS}/2^2$ （2.5 MHz）
0	1	1	$f_{PRS}/2^3$ （1.25 MHz）
1	0	0	$f_{PRS}/2^4$ （625 kHz）
1	0	1	$f_{PRS}/2^5$ （312.5 kHz）
1	1	0	
1	1	1	

注 **1.** 如果外设硬件时钟（f_{PRS}）运行于高速系统时钟（f_{XH}）（XSEL ＝ 1），根据供电电压的不同，f_{PRS} 的工作频率也不同。

- V_{DD} = 2.7 至 5.5 V： $f_{PRS} \leq 10$ MHz
- V_{DD} = 1.8 至 2.7 V： $f_{PRS} \leq 5$ MHz

2. 如果外部硬件时钟 （f_{PRS}）运行于内部高速振荡时钟（f_{RH}）（XSEL = 0），当 1.8 V ≤ V_{DD} < 2.7 V 时，禁止设定 MC0CKS2 = MC0CKS1 = MC0CKS0 = 0（基准时钟： f_{PRS}）。

注意事项 在重写 MC0CKS2 至 MC0CKS0 位之前，将 MC0CTL0 寄存器的第 7 位（MC0PWR）清除为 0。

备注 **1.** f_{PRS}： 外设硬件时钟频率

2. 括号内的数字适用于 f_{PRS} = 10 MHz。

（c） MCG 控制寄存器 2 （MC0CTL2）

该寄存器用于设置发送波特率。

可以通过 8 位存储器操作指令来设置该寄存器。

复位信号的发生会将该寄存器设置为 1FH。

地址： FF4EH　复位后： 1FH　R/W

符号	7	6	5	4	3	2	1	0
MC0CTL2	0	0	0	MC0BRS4	MC0BRS3	MC0BRS2	MC0BRS1	MC0BRS0

MC0BRS4	MC0BRS3	MC0BRS2	MC0BRS1	MC0BRS0	k	5-位计数器的输出时钟选择
0	0	0	×	×	4	$f_{XCLK}/4$
0	0	1	0	0	4	$f_{XCLK}/4$
0	0	1	0	1	5	$f_{XCLK}/5$
0	0	1	1	0	6	$f_{XCLK}/6$
0	0	1	1	1	7	$f_{XCLK}/7$
• • • • •	• • • • •	• • • • •	• • • • •	• • • • •	• • • • •	• • • • •
1	1	1	0	0	28	$f_{XCLK}/28$
1	1	1	0	1	29	$f_{XCLK}/29$
1	1	1	1	0	30	$f_{XCLK}/30$
1	1	1	1	1	31	$f_{XCLK}/31$

注意事项 1. **在重写 MC0BRS4 至 MC0BRS0 位之前，将 MC0CTL0 寄存器的第 7 位（MC0PWR）清除为 0。**

2. 5 位计数器的输出时钟除 2 得到的值是波特率值。

备注 1. f_{XCLK}：　由 MC0CTL1 寄存器的 MC0CKS2 至 MC0CKS0 位选择的基准时钟的频率。

2. k：　由 MC0BRS4 至 MC0BRS0 位设置的值（k = 4，5，6，7，….，31）

3. ×：　无须理会

<1> 波特率

波特率可通过如下表达式计算。

- 波特率　$= \frac{f_{XCLK}}{2 \times k}$ [bps]

f_{XCLK}：　由 MC0CTL1 寄存器的 MC0CKS2 至 MC0CKS0 位选择的基准时钟的频率。

k：　由 MC0CTL2 寄存器的 MC0BRS4 至 MC0BRS0 位设置的值（k = 4，5，6，...，31）

<2> 波特率的误差

波特率误差可通过下面的表达式计算。

- 误差（%）$= \left(\dfrac{\text{实际波特率 (有误差的波特率)}}{\text{理想波特率 (正确的波特率)}} - 1 \right) \times 100\ [\%]$

注意事项　在发送期间，必须保持波特率误差在接收目的方允许误差范围内。

例：　基准时钟的频率 = 2.5 MHz = 2,500,000 Hz

MC0CTL2 寄存器的 MC0BRS4 至 MC0BRS0 位的设置值= 10000B （k = 16）

目标波特率 = 76,800 bps

波特率 = 2.5 M/（2 × 16）

= 2,500,000/（2 × 16） = 78125 [bps]

误差= （78,125/76,800 – 1） × 100

= 1.725 [%]

<3> 波特率设置的示例

波特率 [bps]	f_{PRS} = 10.0 MHz				f_{PRS} = 8.38 MHz				f_{PRS} = 8.0 MHz				f_{PRS} = 6.0 MHz			
	MC0CKS2 至 MC0CKS0	k	计算值	ERR [%]	MC0CKS2 至 MC0CKS0	k	计算值	ERR [%]	MC0CKS2 至 MC0CKS0	k	计算值	ERR [%]	MC0CKS2 至 MC0CKS0	k	计算值	ERR [%]
4800	–	–	–	–	5，6 或 7	27	4850	1.03	5，6 或 7	26	4808	0.16	5，6 或 7	20	4688	–2.34
9600	5，6，或 7	16	9766	1.73	4	27	9699	1.03	5，6，或 7	13	9615	0.16	4	20	9375	–2.34
19200	5	8	19531	1.73	3	27	19398	1.03	4	13	19231	0.16	4	10	19750	–2.34
31250	4	10	31250	0	2	17	30809	–1.41	4	8	31250	0	2	24	31250	0
38400	4	8	39063	1.73	2	27	38796	1.03	3	13	38462	0.16	2	20	37500	–2.34
56000	3	11	56819	1.46	2	19	55132	–1.55	3	9	55556	–0.79	1	27	55556	–0.79
62500	2	20	62500	0	2	17	61619	–1.41	3	8	62500	0	2	12	62500	0
76800	2	16	78125	1.73	1	27	77592	1.03	2	13	76923	0.16	2	10	75000	–2.34
115200	1	22	113636	–1.36	2	9	116389	1.03	1	17	117647	2.12	1	13	115385	0.16
125000	1	20	125000	0	1	17	123235	–1.41	1	16	125000	0	1	12	125000	0
153600	1	16	156250	1.73	2	7	149643	–2.58	1	13	153846	0.16	1	10	150000	–2.34
250000	1	10	250000	0	1	8	261975	4.75	1	8	250000	0	1	6	250000	0
					0	17	246471	–1.41								

备注　MC0CKS2 至 MC0CKS0：　MCG 控制寄存器 1（MC0CTL1）的第 2 位至第 0 位（基准时钟（f_{XCLK}）的设置）

k：　由 MCG 控制寄存器 2（MC0CTL2）第 4 位至第 0 位（MC0BRS4 至 MC0BRS0）设置的值（k = 4，5，6，…，31）

f_{PRS}：　外设硬件时钟频率

ERR：　波特率误差

（d） 端口模式寄存器 3（PM3）

该寄存器按位设置端口 3 输入/输出模式。

当 P32/TOH0/MCGO 引脚用于曼彻斯特编码输出时，清除 PM32 为 0，并且清除 P32 的输出锁存器为 0。

可以通过 1 位或 8 位存储器操作指令来设置 PM3。

复位信号的发生会将该寄存器设置为 FFH。

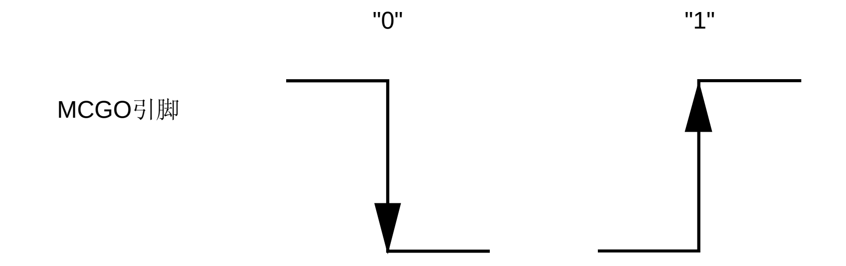
地址： FF23H　复位后： FFH　R/W

符号	7	6	5	4	3	2	1	0
PM3	1	1	1	PM34	PM33	PM32	PM31	1

PM3n	P3n 引脚 I/O 模式选择 （n = 1 至 4）
0	输出模式（输出缓冲器打开）
1	输入模式（输出缓冲器关闭）

（2）　曼彻斯特编码输出的"0"和"1"的格式

在 78K0/LE3 中，曼彻斯特编码输出的"0" 和"1"的格式如下。



（3）　发送操作

在曼彻斯特编码发生器模式下，数据按 1 至 8 位单元发送。数据位按照曼彻斯特编码格式发送。如果 MCG 控制寄存器 0（MC0CTL0）的第 7 位（MC0PWR）设置为 1，则使能发送。

通过 MC0CTL0 寄存器的第 0 位（MC0OLV），可以设置发送暂停时的输出值。

发送数据位长度设置到 MCG 发送位计数规范寄存器（MC0BIT）之后，通过写入一个值到 MCG 发送缓冲器寄存器（MC0TX），启动发送。在发送开始时序，MC0BIT 的值被传送到 3 位计数器，MC0TX 的数据被传送到 8 位移位寄存器。当 MC0TX 值被传送到 8 位移位寄存器时，发生一个中断请求信号（INTMCG）。8 位移位寄存器通过波特率时钟不断移位，并且与波特率时钟异或后的信号从 MCGO 引脚输出。

执行连续发送时，INTMCG 发生后，在数据发送期间将下一个数据设置到 MC0BIT 和 MC0TX 。

要连续发送，下一个发送数据写入到 MC0TX 的操作必须在图 18-8 中（3）和（4）期间内完成。连续发送期间，在写 MC0TX 之前，重写 MC0BIT。

图 **18-8.** 曼彻斯特编码发生器模式的时序（**LSB** 先行）（**1/4**）

（**1**）　发送时序（**MC0OLV = 1**，总共发送位长度 **= 8** 位）

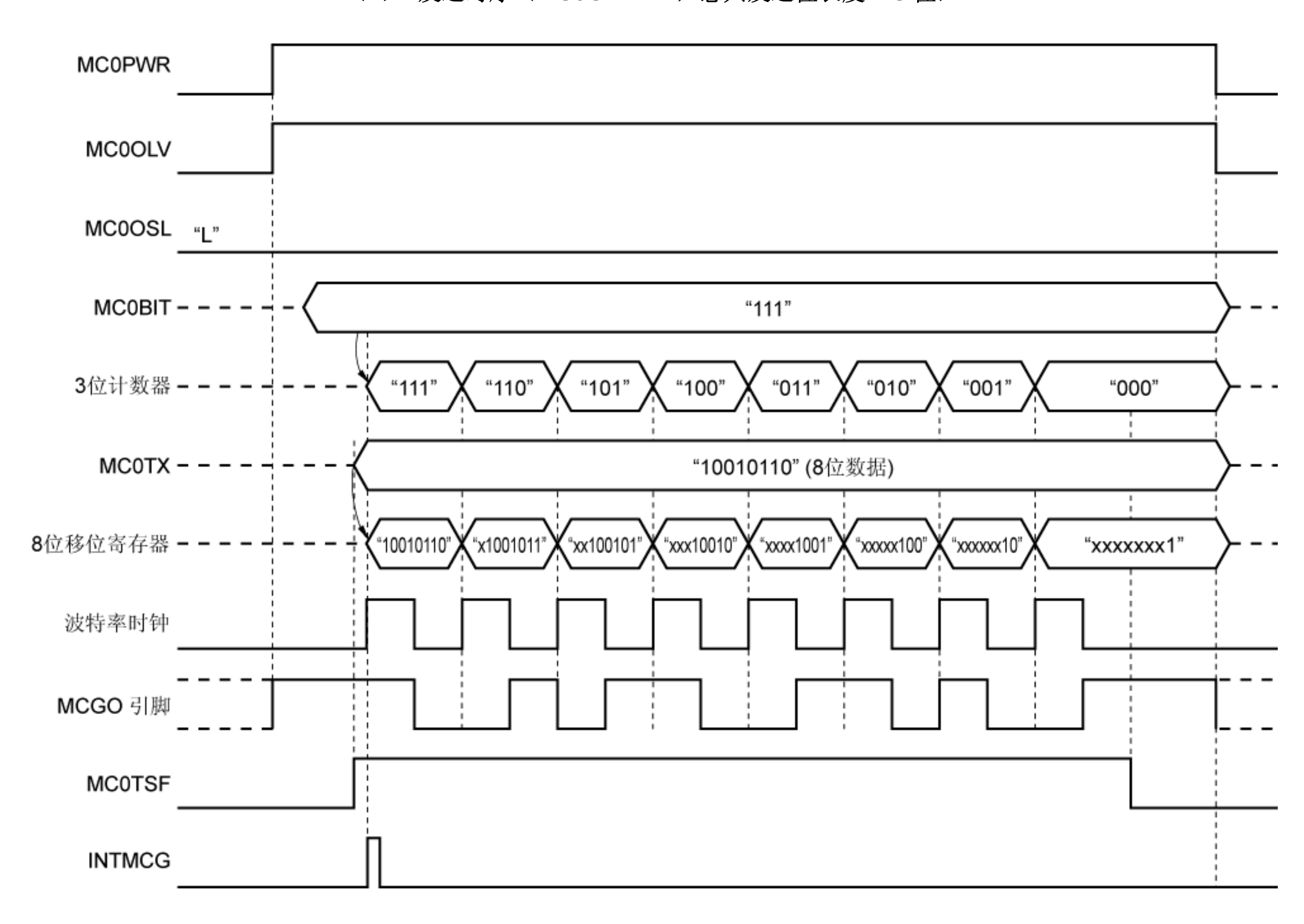


图 **18-8.** 曼彻斯特编码发生器模式的时序（**LSB** 先行）（**2/4**）

（**2**）　发送时序 （**MC0OLV = 0**，总共发送位长度 **= 8** 位）

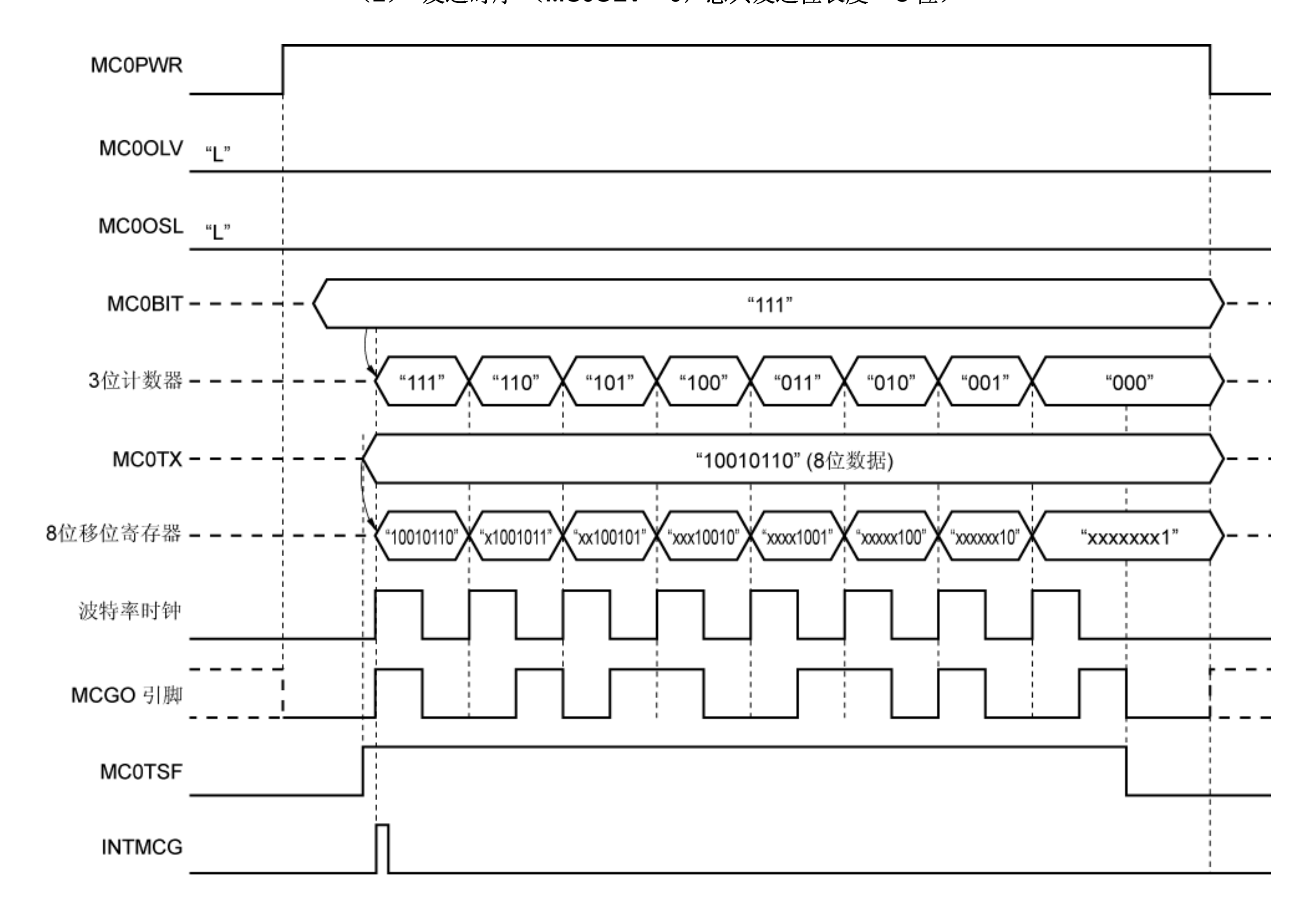
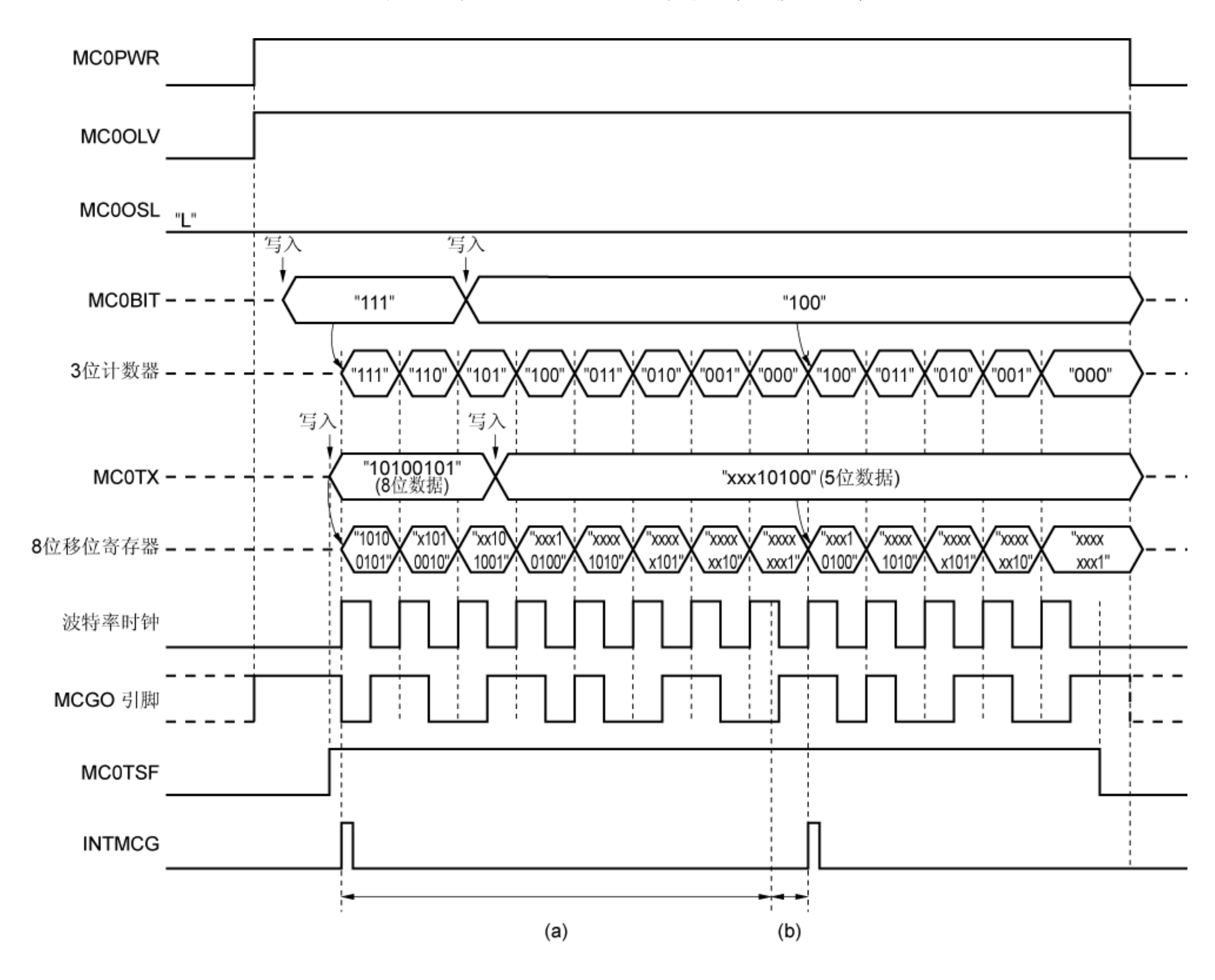


图 **18-8.** 曼彻斯特编码发生器模式的时序（**LSB** 先行）（**3/4**）

（**3**）　发送时序（**MC0OLV = 1**，总共发送位长度 **= 13** 位）



（a）：　　“8-位传输时期” –（b）

（b）：　　“波特率的 1/2 周期” + 发送数据最后一位之前的 1 个时钟（f_{XCLK}）

f_{XCLK}：　　使用 MC0CTL1 寄存器的 MC0CKS2 至 MC0CKS0 位选择操作基准时钟的频率

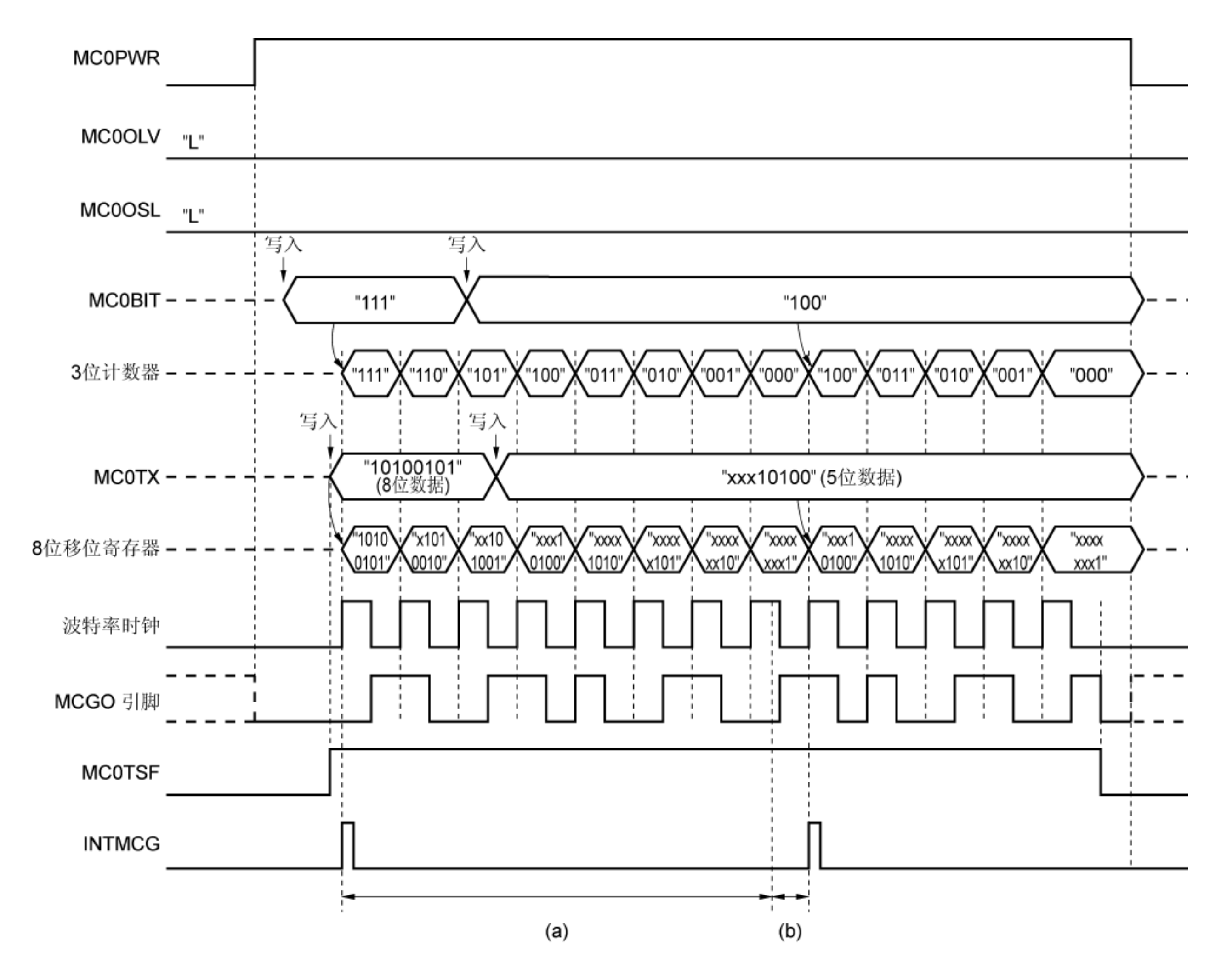
最后一位：　当 3 位计数器= 000 时的传输位

注意事项　连续发送期间，下一个发送数据写入到 **MC0TX** 的操作必须在（**a**）时期内完成。如果下一个发送数据写入到 **MC0TX** 的操作在（**b**）时期内执行，最后一位被发送后经过 **2** 个时钟（f_{XCLK}），开始下一次数据发送。

连续发送期间，写入 **MC0TX** 之前，重写 **MC0BIT**。

图 18-8. 曼彻斯特编码发生器模式的时序（LSB 先行）（4/4）

（4）　发送时序（MC0OLV = 0，总共发送位长度 = 13 位）



（a）：　“8-位传输时期” –（b）
（b）：　“波特率的 1/2 周期” + 发送数据最后一位之前的 1 个时钟（f_{XCLK}）
f_{XCLK}：　使用 MC0CTL1 寄存器的 MC0CKS2 至 MC0CKS0 位选择操作基准时钟的频率
最后一位：　当 3 位计数器= 000 时的传输位

注意事项　连续发送期间，下一个发送数据写入到 **MC0TX** 的操作必须在（**a**）时期内完成。如果下一个发送数据写入到 **MC0TX** 的操作在（**b**）时期内执行，最后一位被发送后经过 **2** 个时钟（f_{XCLK}），开始下一次数据发送。
连续发送期间，写入 **MC0TX** 之前，重写 **MC0BIT**。

18.4.3 位序列缓冲器模式

位序列缓冲器模式用于使用 MCGO 引脚输出序列信号。

（1）　寄存器描述

MCG 控制寄存器 0（MC0CTL0），MCG 控制寄存器 1（MC0CTL1），和 MCG 控制寄存器 2（MC0CTL2）用于设置位序列缓冲器模式。

（a） MCG 控制寄存器 0 （MC0CTL0）

可以通过 1 位或 8 位存储器操作指令来设置该寄存器。

复位信号的发生会将该寄存器设置为 10H。

地址： FF4CH　复位后： 10H　R/W

符号	<7>	6	5	<4>	3	2	<1>	<0>
MC0CTL0	MC0PWR	0	0	MC0DIR	0	0	MC0OSL	MC0OLV

MC0PWR	操作控制
0	操作停止
1	操作允许

MC0DIR	首位规范
0	MSB
1	LSB

MC0OSL	数据格式
0	曼彻斯特编码
1	位序列数据

MC0OLV	发送暂停时的输出电平
0	低电平
1	高电平

注意事项　在重写 MC0DIR，MC0OSL 和 MC0OLV 位（当 MC0PWR 位被置位（1）时的同时，可以通过 8 位存储器操作指令重写这些位）之前，清除（0）MC0PWR 位。

（b） MCG 控制寄存器 1 （MC0CTL1）

该寄存器用于设置曼彻斯特编码发生器的基准时钟。

可以通过 8 位存储器操作指令来设置该寄存器。

复位信号的发生会将该寄存器清除为 00H。

地址： FF4DH　复位后： 00H　R/W

符号	7	6	5	4	3	2	1	0
MC0CTL1	0	0	0	0	0	MC0CKS2	MC0CKS1	MC0CKS0

MC0CKS2	MC0CKS1	MC0CKS0	基准时钟（f_{XCLK}）选择
0	0	0	f_{PRS} （10 MHz）
0	0	1	$f_{PRS}/2$ （5 MHz）
0	1	0	$f_{PRS}/2^2$ （2.5 MHz）
0	1	1	$f_{PRS}/2^3$ （1.25 MHz）
1	0	0	$f_{PRS}/2^4$ （625 kHz）
1	0	1	$f_{PRS}/2^5$ （312.5 kHz）
1	1	0	
1	1	1	

注意事项 **在重写 MC0CKS2 至 MC0CKS0 位之前，将 MC0CTL0 寄存器的第 7 位（MC0PWR）清除为 0。**

备注 **1.** f_{PRS}： 外设硬件时钟频率

2. 括号内的数字适用于 f_{PRS} = 10 MHz。

（c） MCG 控制寄存器 2 （MC0CTL2）

该寄存器用于设置发送波特率。

可以通过 8 位存储器操作指令来设置该寄存器。

复位信号的发生会将该寄存器设置为 1FH。

地址： FF4EH　复位后： 1FH　R/W

符号	7	6	5	4	3	2	1	0
MC0CTL2	0	0	0	MC0BRS4	MC0BRS3	MC0BRS2	MC0BRS1	MC0BRS0

MC0BRS4	MC0BRS3	MC0BRS2	MC0BRS1	MC0BRS0	k	5 位计数器输出时钟的选择
0	0	0	×	×	4	$f_{XCLK}/4$
0	0	1	0	0	4	$f_{XCLK}/4$
0	0	1	0	1	5	$f_{XCLK}/5$
0	0	1	1	0	6	$f_{XCLK}/6$
0	0	1	1	1	7	$f_{XCLK}/7$
•	•	•	•	•	•	•
1	1	1	0	0	28	$f_{XCLK}/28$
1	1	1	0	1	29	$f_{XCLK}/29$
1	1	1	1	0	30	$f_{XCLK}/30$
1	1	1	1	1	31	$f_{XCLK}/31$

注意事项 1. 在重写 MC0BRS4 至 MC0BRS0 位之前，将 MC0CTL0 寄存器的第 7 位（MC0PWR）清除为 0。

2. 5 位计数器的输出时钟除 2 得到的值是波特率值。

备注 **1.** f_{XCLK}：　由 MC0CTL1 寄存器的 MC0CKS2 至 MC0CKS0 位选择的基准时钟的频率。

2. k：　由 MC0BRS4 至 MC0BRS0 位设置的值（k = 4，5，6，7，....，31）

3. ×：　无须理会

<1> 波特率

波特率可通过如下表达式计算。

- 波特率 $= \frac{f_{XCLK}}{2 \times k}$ [bps]

f_{XCLK}：　由 MC0CTL1 寄存器的 MC0CKS2 至 MC0CKS0 位选择的基准时钟的频率。

k：　由 MC0CTL2 寄存器的 MC0BRS4 至 MC0BRS0 位设置的值（k = 4，5，6，...，31）

<2> 波特率误差

波特率误差可通过下面的表达式计算。

- 误差（%）= $\left(\dfrac{\text{实际波特率 (有误差的波特率)}}{\text{理想波特率 (正确的波特率)}} - 1\right) \times 100$ [%]

注意事项　在发送期间，必须保持波特率误差在接收目的方允许误差范围内。

例：　基准时钟的频率= 2.5 MHz = 2,500,000 Hz

MC0CTL2 寄存器的 MC0BRS4 至 MC0BRS0 位的设置值= 10000B （k = 16）

目标波特率 = 76,800 bps

波特率 = 2.5 M/（2 × 16）

= 2,500,000/（2 × 16） = 78125 [bps]

误差 = （78,125/76,800 – 1） × 100

= 1.725 [%]

<3> 波特率设置的示例

波特率 [bps]	f_{PRS} = 10.0 MHz				f_{PRS} = 8.38 MHz				f_{PRS} = 8.0 MHz				f_{PRS} = 6.0 MHz			
	MC0CKS2 至 MC0CKS0	k	计数值	ERR [%]	MC0CKS2 至 MC0CKS0	k	计数值	ERR [%]	MC0CKS2 至 MC0CKS0	k	计数值	ERR [%]	MC0CKS2 至 MC0CKS0	k	计数值	ERR [%]
4800	–	–	–	–	5，6 或 7	27	4850	1.03	5，6 或 7	26	4808	0.16	5，6 或 7	20	4688	–2.34
9600	5，6 或 7	16	9766	1.73	4	27	9699	1.03	5，6 或 7	13	9615	0.16	4	20	9375	–2.34
19200	5	8	19531	1.73	3	27	19398	1.03	4	13	19231	0.16	4	10	19750	–2.34
31250	4	10	31250	0	2	17	30809	–1.41	4	8	31250	0	2	24	31250	0
38400	4	8	39063	1.73	2	27	38796	1.03	3	13	38462	0.16	2	20	37500	–2.34
56000	3	11	56819	1.46	2	19	55132	–1.55	3	9	55556	–0.79	1	27	55556	–0.79
62500	2	20	62500	0	2	17	61619	–1.41	3	8	62500	0	2	12	62500	0
76800	2	16	78125	1.73	1	27	77592	1.03	2	13	76923	0.16	2	10	75000	–2.34
115200	1	22	113636	–1.36	2	9	116389	1.03	1	17	117647	2.12	1	13	115385	0.16
125000	1	20	125000	0	1	17	123235	–1.41	1	16	125000	0	1	12	125000	0
153600	1	16	156250	1.73	2	7	149643	–2.58	1	13	153846	0.16	1	10	150000	–2.34
250000	1	10	250000	0	1	8	261975	4.75	1	8	250000	0	1	6	250000	0
					0	17	246471	–1.41								

备注　MC0CKS2 至 MC0CKS0：　MCG 控制寄存器 1（MC0CTL1）的第 2 位至第 0 位（基准时钟（f_{XCLK}）的设置）

k：　由 MCG 控制寄存器 2（MC0CTL2）第 4 位至第 0 位（MC0BRS4 至 MC0BRS0）设置的值（k = 4，5，6，…，31）

f_{PRS}：　外设硬件时钟频率

ERR：　波特率误差

（d） 端口模式寄存器 3（PM3）

该寄存器按位设置端口 3 输入/输出模式。

当 P32/TOH0/MCGO 引脚用于位序列数据输出时，清除 PM32 为 0，并且清除 P32 的输出锁存器为 0。

可以通过 1 位或 8 位存储器操作指令来设置 PM3。

复位信号的发生会将该寄存器设置为 FFH。

地址： FF23H 复位后： FFH R/W

符号	7	6	5	4	3	2	1	0
PM3	1	1	1	PM34	PM33	PM32	PM31	1

PM3n	P3n 引脚 I/O 模式选择 （n = 1 至 4）
0	输出模式（输出缓冲器打开）
1	输入模式（输出缓冲器关闭）

（2） 发送操作

在位序列缓冲器模式下，数据按 1 至 8 位单元发送。如果 MCG 控制寄存器 0（MC0CTL0）的第 7 位（MC0PWR）设置为 1，则使能发送。

通过 MC0CTL0 寄存器的第 0 位（MC0OLV），可以设置发送暂停时的输出值。

发送数据位长度设置到 MCG 发送位计数规范寄存器（MC0BIT）之后，通过写入一个值到 MCG 发送缓冲器寄存器（MC0TX），启动发送。在发送开始时序，MC0BIT 的值被传送到 3 位计数器，MC0TX 的数据被传送到 8 位移位寄存器。当 MC0TX 值被传送到 8 位移位寄存器时，发生一个中断请求信号（INTMCG）。8 位移位寄存器通过波特率时钟不断移位，并且从 MCGO 引脚输出。

执行连续发送时，INTMCG 发生后，在数据发送期间将下一个数据设置到 MC0BIT 和 MC0TX 。

要连续发送，下一个发送数据写入到 MC0TX 的操作必须在图 18-8 中（3）和（4）期间内完成。连续发送期间，在写 MC0TX 之前，重写 MC0BIT。

图 18-9. 位序列缓冲器模式的时序（LSB 先行）（1/4）

（1） 发送时序（MC0OLV = 1，总共发送位长度 = 8 位）

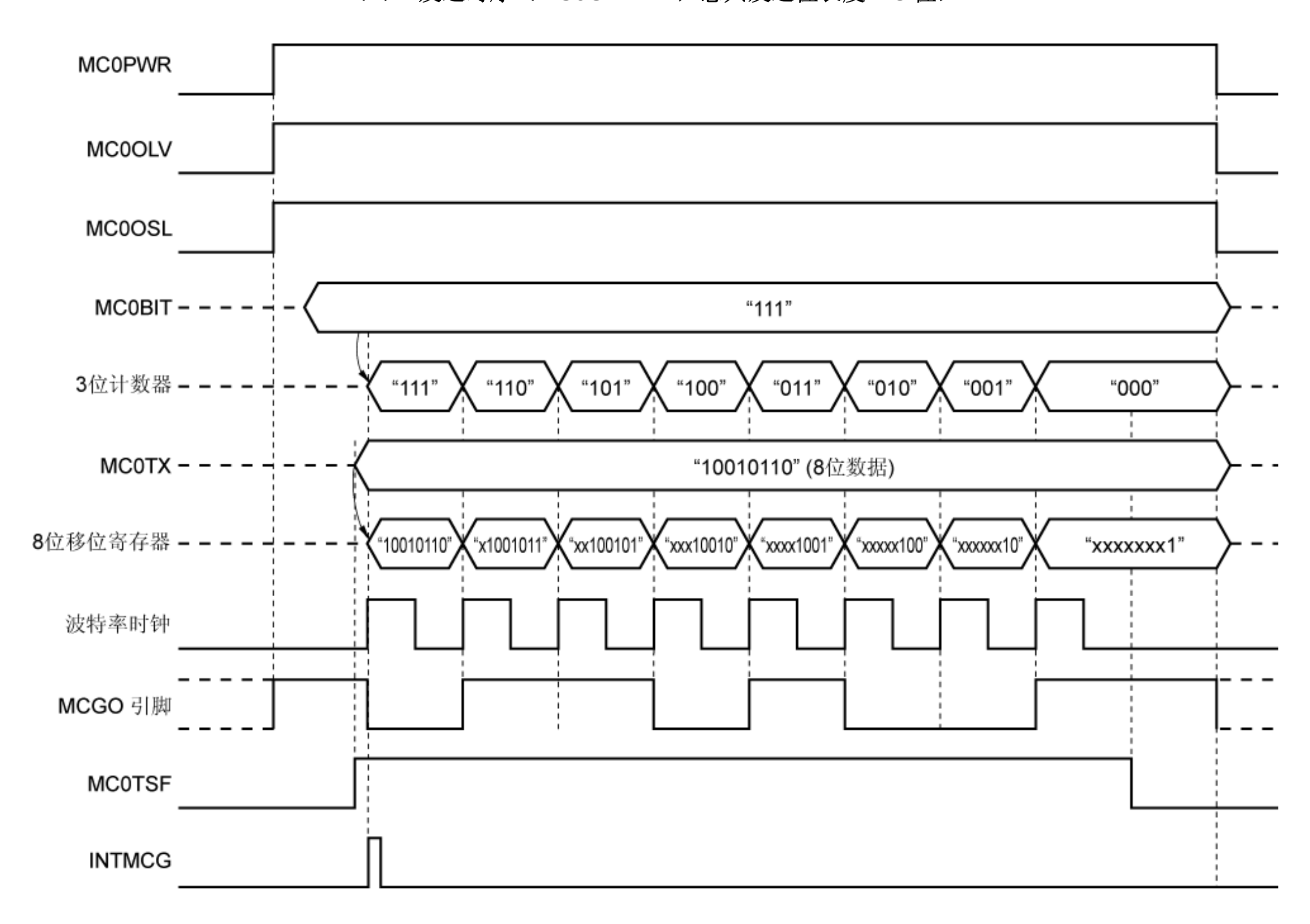


图 **18-9.** 位序列缓冲器模式的时序（**LSB** 先行）（**2/4**）

（**2**）　发送时序（**MC0OLV = 0**，总共发送位长度 **= 8** 位）

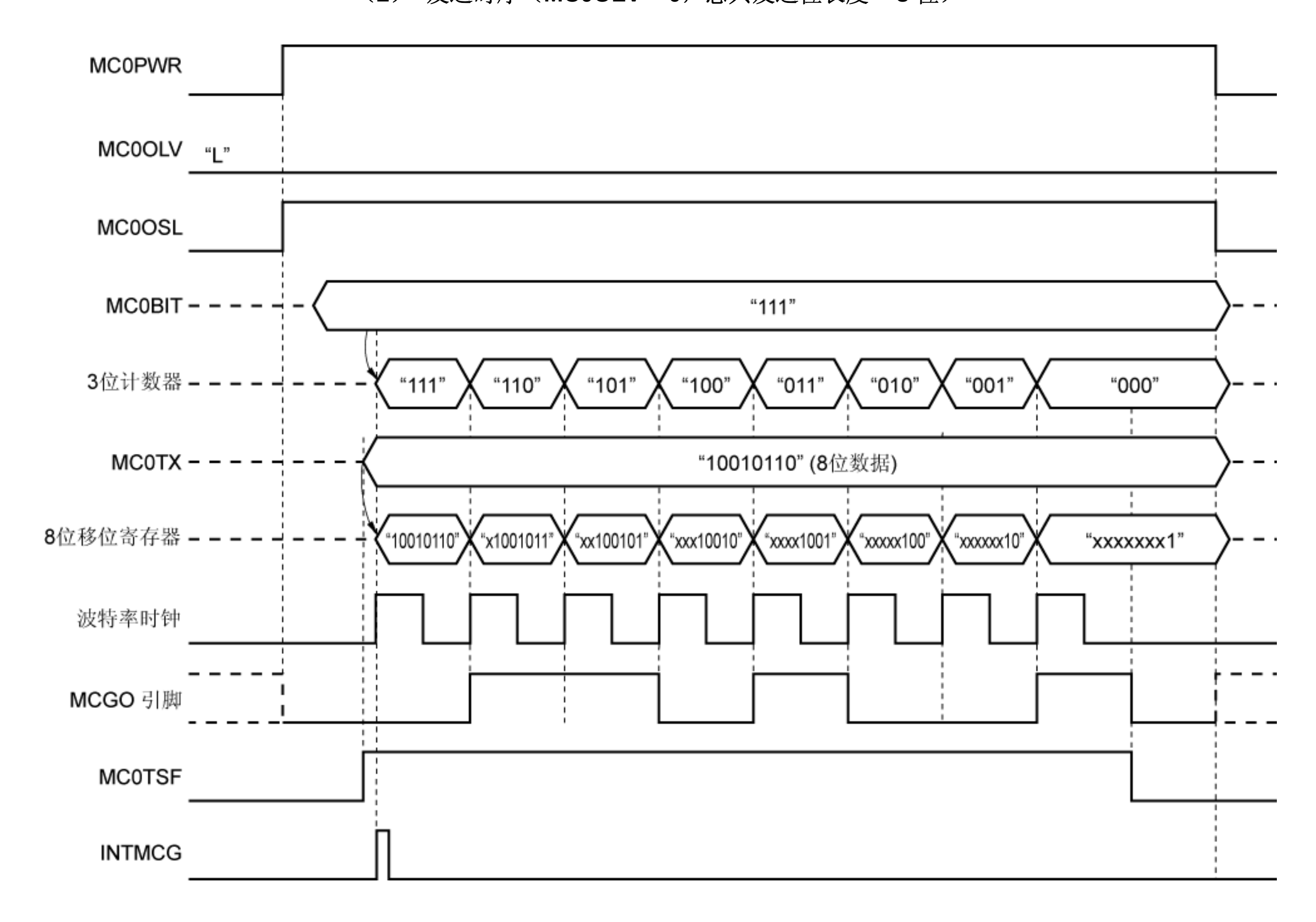
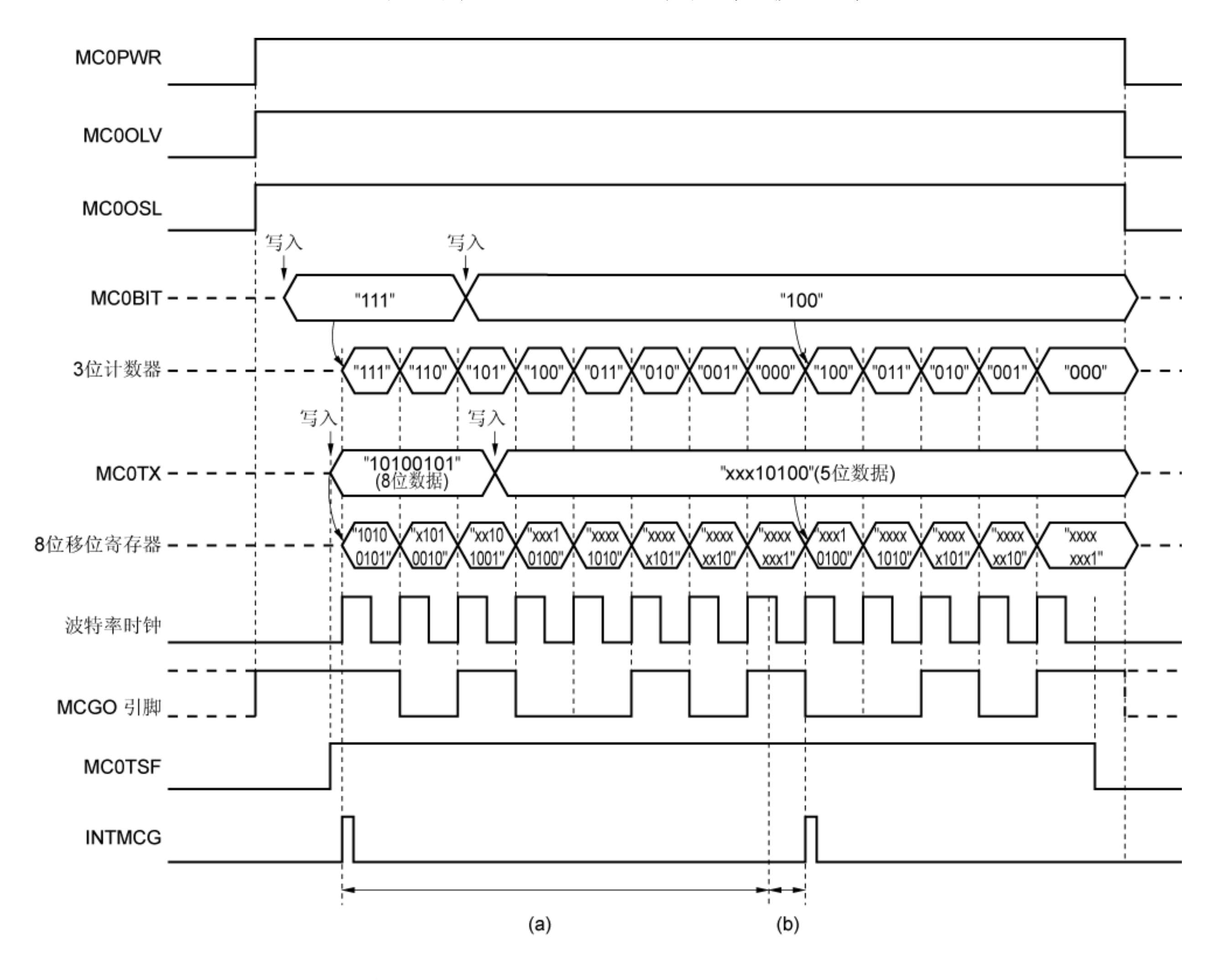


图 **18-9.** 位序列缓冲器模式的时序（**LSB** 先行）（**3/4**）

（**3**） 发送时序（**MC0OLV = 1**，总共发送位长度 **= 13** 位）



（a）： “8-位传输时期” – （b）

（b）： “波特率的 1/2 周期” + 发送数据最后一位之前的 1 个时钟（f_{XCLK}）

f_{XCLK}： 使用 MC0CTL1 寄存器的 MC0CKS2 至 MC0CKS0 位选择操作基准时钟的频率

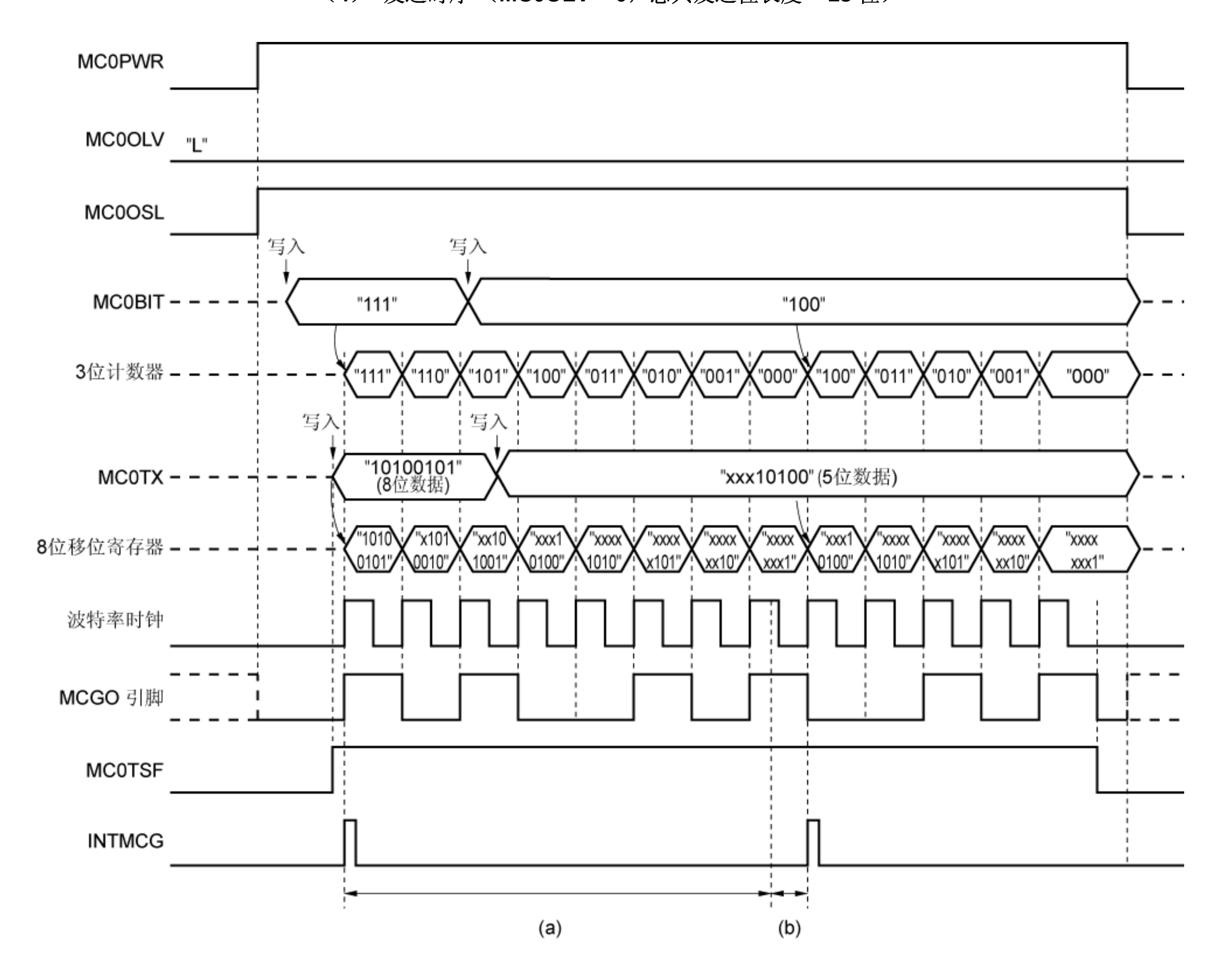
最后一位： 当 3 位计数器= 000 时的传输位

注意事项 连续发送期间，下一个发送数据写入到 **MC0TX** 的操作必须在（**a**）时期内完成。如果下一个发送数据写入到 **MC0TX** 的操作在（**b**）时期内执行，最后一位被发送后经过 **2** 个时钟（f_{XCLK}），开始下一次数据发送。

连续发送期间，写入 **MC0TX** 之前，重写 **MC0BIT**。

图 **18-9.** 位序列缓冲器模式的时序 （**LSB** 先行） （**4/4**）

（**4**） 发送时序 （**MC0OLV = 0**，总共发送位长度 **= 13** 位）



（a）：　“8-位传输时期” –（b）

（b）：　“波特率的 1/2 周期” + 发送数据最后一位之前的 1 个时钟（f_{XCLK}）

f_{XCLK}：　使用 MC0CTL1 寄存器的 MC0CKS2 至 MC0CKS0 位选择操作基准时钟的频率

最后一位：　当 3 位计数器= 000 时的传输位

注意事项　连续发送期间，下一个发送数据写入到 **MC0TX** 的操作必须在（**a**）时期内完成。如果下一个发送数据写入到 **MC0TX** 的操作在（**b**）时期内执行，最后一位被发送后经过 **2** 个时钟（f_{XCLK}），开始下一次数据发送。

连续发送期间，写入 **MC0TX** 之前，重写 **MC0BIT**。

第十九章　遥控器接收器

19.1 遥控器接收器功能

遥控器接收器使用如下遥控器模式。

- 类型 A 接收模式 … 提供引导脉冲（半个时钟）
- 类型 B 接收模式 … 提供引导脉冲（一个时钟）
- 类型 C 接收模式 … 不提供引导脉冲

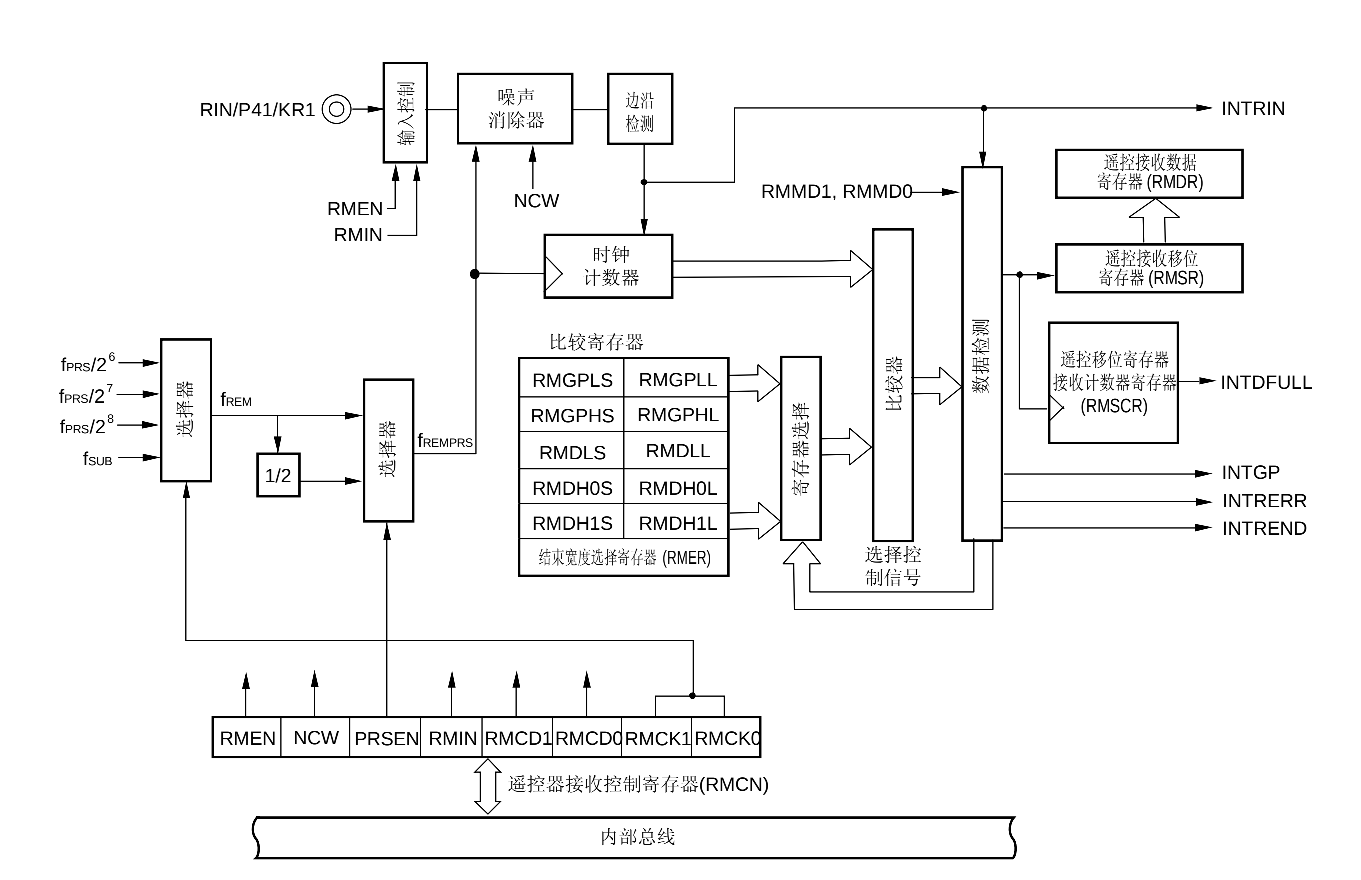
19.2 遥控器接收器配置

遥控器接收器包括如下硬件。

表 19-1. 遥控器接收器配置

项目	配置
寄存器	遥控器接收移位寄存器（RMSR） 遥控器接收数据寄存器（RMDR） 遥控器移位寄存器接收计数器寄存器（RMSCR） 遥控器接收 GPLS 比较寄存器（RMGPLS） 遥控器接收 GPLL 比较寄存器（RMGPLL） 遥控器接收 GPHS 比较寄存器（RMGPHS） 遥控器接收 GPHL 比较寄存器（RMGPHL） 遥控器接收 DLS 比较寄存器（RMDLS） 遥控器接收 DLL 比较寄存器（RMDLL） 遥控器接收 DH0S 比较寄存器（RMDH0S） 遥控器接收 DH0L 比较寄存器（RMDH0L） 遥控器接收 DH1S 比较寄存器（RMDH1S） 遥控器接收 DH1L 比较寄存器（RMDH1L） 遥控器接收结束宽度选择寄存器（RMER）
控制寄存器	遥控器接收中断状态寄存器（INTS） 遥控器接收中断状态清除寄存器（INTC） 遥控器接收控制寄存器（RMCN）

图 **19-1.** 遥控器接收器的框图



（1）遥控器接收移位寄存器（RMSR）

这是个 8 位寄存器，用于遥控器数据的接收。

数据首先存储在第 7 位。每次存储新数据时，已存储数据向低位移位。因此，最新数据存储在第 7 位，第一个数据存储在第 0 位。

可以通过 8 位存储器操作指令来读取 RMSR。

复位信号的发生会将 RMSR 设置为 00H。

同样，在下列任一条件下 RMSR 被清除为 00H。

- 遥控器停止操作（RMEN = 0）。
- 检测到错误（产生 INTRERR）。
- 产生 INTDFULL。
- 在产生 INTREND 后读取 RMSR。

注意事项 遥控器接收期间，禁止读取 RMSR。完成接收后，然后读取 RMSR。当读取操作完成时，RMSR 被清除。因此，值一旦被读取，就无法保证。

（2）遥控器接收数据寄存器（RMDR）

该寄存器用于保存遥控器接收数据。当遥控器接收移位寄存器（RMSR）溢出时，RMSR 中的数据被传送到 RMDR。第 7 位存储最后的数据，第 0 位存储第一个数据。数据从 RMSR 传送到 RMDR 的同时，产生 INTDFULL。

可以通过 8 位存储器操作指令来读取 RMDR。

复位信号的发生会将 RMDR 设置为 00H。

禁止遥控器操作时（RMEN = 0），RMDR 被清除为 00H。

注意事项　当产生 INTDFULL 后，在接收到下一个 8 位数据之前读取 RMDR。如果在读操作完成之前，产生下一个 INTDFULL，RMDR 被覆盖。

（3）遥控器移位寄存器接收计数器寄存器（RMSCR）

这是一个 3 位计数器寄存器，用于在遥控器接收完成（产生 INTREND）时表示保存在遥控器接收移位寄存器（RMSR）中有效位的数量。读取该寄存器的值可以确认位的数量，即使接收到的数据格式不是 8 位的整数倍。

可以通过 8 位存储器操作指令来读取 RMSCR。

复位信号的发生会将 RMSCR 设置为 00H。

在下列任一条件下 RMSCR 被清除为 00H。

- 遥控器停止操作（RMEN = 0）
- 检测到错误（产生 INTRERR）
- 在产生 INTREND 后读取 RMSR

注意事项　当 INTREND 产生后，紧邻在读取 RMSR 之前，进行 RMSCR 的读取。如果读取发生在其他时序，则值无法保证。

图 19-2. RMSR，RMSCR 和 RMDR 寄存器的操作示例
当接收 1010101011111111B（16 位）时

	RMSR								RMSCR	RMDR
	7	6	5	4	3	2	1	0		
复位后	0	0	0	0	0	0	0	0	00H	00000000B
接收 1 位	1	0	0	0	0	0	0	0	01H	00000000B
接收 2 位	0	1	0	0	0	0	0	0	02H	00000000B
接收 3 位	1	0	1	0	0	0	0	0	03H	00000000B
…	…	…	…	…	…	…	…	…	…	…
接收 7 位	1	0	1	0	1	0	1	0	07H	00000000B
接收 8 位 ↓ RMDR 传输	0 ↓ 0	1 ↓ 0	0 ↓ 0	1 ↓ 0	0 ↓ 0	1 ↓ 0	0 ↓ 0	1 ↓ 0	00H ↓ 00H	00000000B ↓ 01010101B
接收 9 位	1	0	0	0	0	0	0	0	01H	01010101B
接收 10 位	1	1	0	0	0	0	0	0	02H	01010101B
…	…	…	…	…	…	…	…	…	…	…
接收 16 位 ↓ RMDR 传输	1 ↓ 0	1 ↓ 0	1 ↓ 0	1 ↓ 0	1 ↓ 0	1 ↓ 0	1 ↓ 0	1 ↓ 0	00H ↓ 00H	01010101B ↓ 11111111B

（4）遥控器接收 GPLS 比较寄存器（RMGPLS）（类型 B 接收模式）

该寄存器用于检测遥控器引导脉冲（短侧）的低电平。

可以通过 8 位存储器操作指令来设置 RMGPLS。

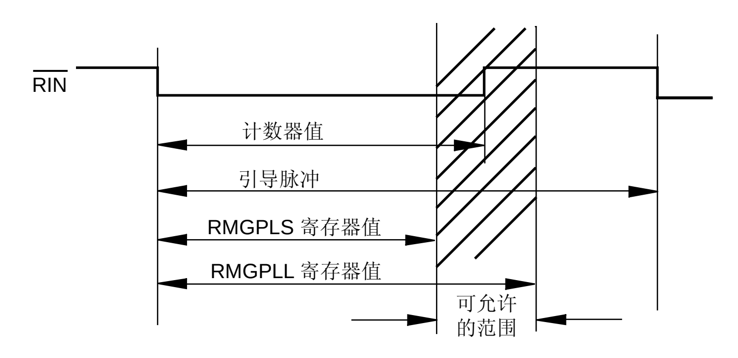
复位信号的发生会将 RMGPLS 设置为 00H。

（5）遥控器接收 GPLL 比较寄存器（RMGPLL）（类型 B 接收模式）

该寄存器用于检测遥控器引导脉冲（长侧）的低电平。

可以通过 8 位存储器操作指令来设置 RMGPLL。

复位信号的发生会将 RMGPLL 设置为 00H。



（6）遥控器接收 GPHS 比较寄存器（RMGPHS）（仅限类型 A，类型 B 接收模式）

该寄存器用于检测遥控器引导脉冲（短侧）的高电平。

可以通过 8 位存储器操作指令来设置 RMGPHS。

复位信号的发生会将 RMGPHS 设置为 00H。

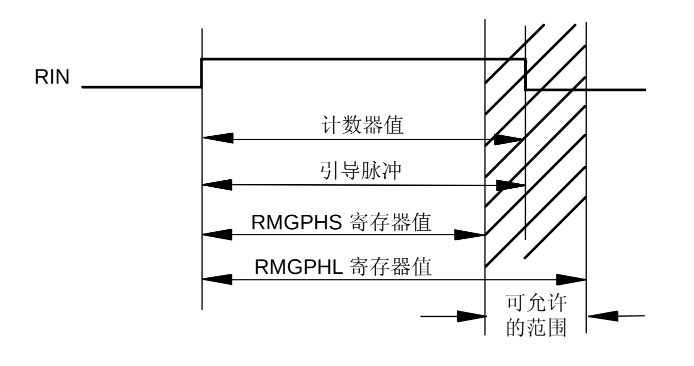
（7）遥控器接收 GPHL 比较寄存器（RMGPHL）（仅限类型 A，类型 B 接收模式）

该寄存器用于检测遥控器引导脉冲（长侧）的高电平。

可以通过 8 位存储器操作指令来设置 RMGPHL。

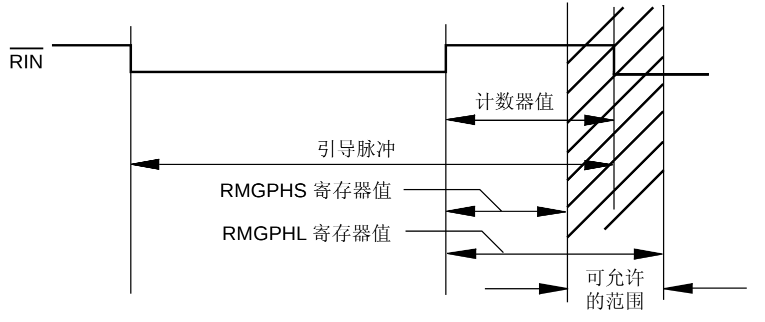
复位信号的发生会将 RMGPHL 设置为 00H。

（a）类型 A 接收模式



如果 RMGPHS ≤ 计数器值 < RMGPHL，
就认为成功接收到引导
脉冲的高电平。

（b）类型 B 接收模式



如果 RMGPHS ≤ 计数器值 < RMGPHL，
就认为成功接收到引导脉冲的高电平 .

（8）遥控器接收 DLS 比较寄存器（RMDLS）

该寄存器用于检测遥控器数据（短侧）的低电平。

可以通过 8 位存储器操作指令来设置 RMDLS。

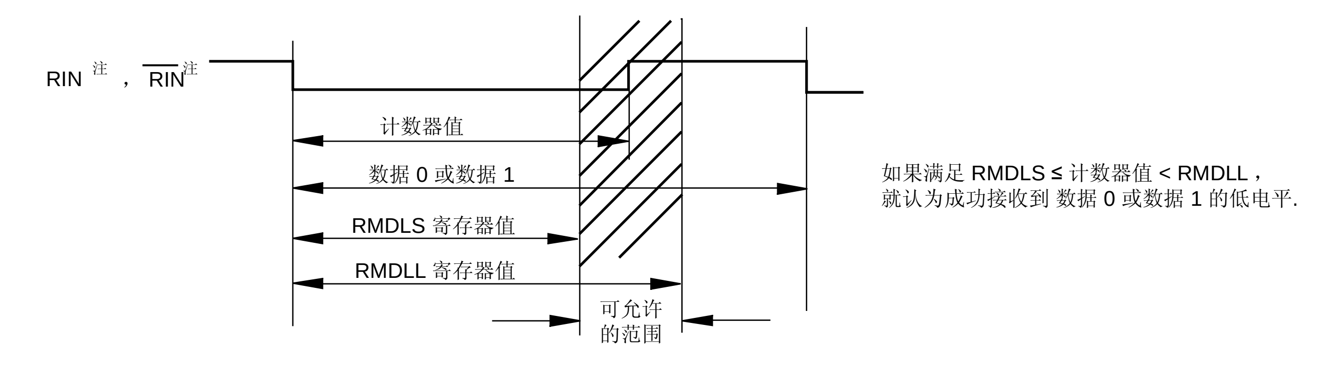
复位信号的发生会将 RMDLS 设置为 00H。

（9）遥控器接收 DLL 比较寄存器（RMDLL）

该寄存器用于检测遥控器数据（长侧）的低电平。

可以通过 8 位存储器操作指令来设置 RMDLL。

复位信号的发生会将 RMDLL 设置为 00H。



注　　在类型 A 接收模式下产生 RIN，在类型 B 和类型 C 接收模式下产生 $\overline{\text{RIN}}$。

（10） 遥控器接收 DH0S 比较寄存器（RMDH0S）

寄存器用于检测遥控器数据 0（短侧）的高电平。

可以通过 8 位存储器操作指令来设置 RMDH0S。

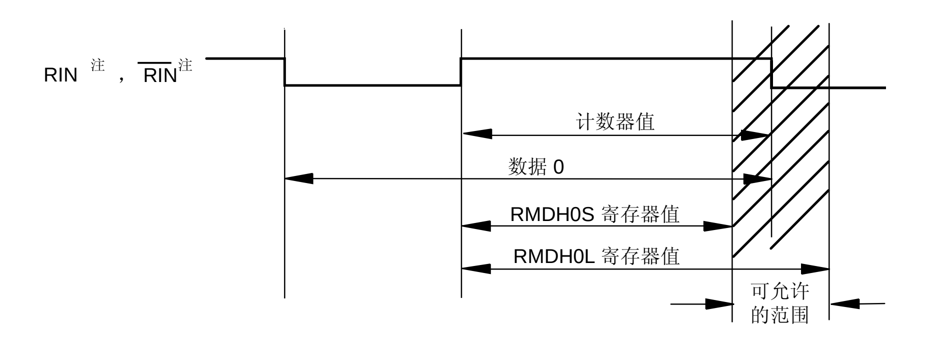
复位信号的发生会将 RMDH0S 设置为 00H。

（11） 遥控器接收 DH0L 比较寄存器（RMDH0L）

寄存器用于检测遥控器数据 0（长侧）的高电平。

可以通过 8 位存储器操作指令来设置 RMDH0L。

复位信号的发生会将 RMDH0L 设置为 00H。



注　　在类型 A 接收模式下产生 RIN，在类型 B 和类型 C 接收模式下产生 $\overline{\text{RIN}}$。

（12） 遥控器接收 DH1S 比较寄存器（RMDH1S）

寄存器用于检测遥控器数据 1（短侧）的高电平。

可以通过 8 位存储器操作指令来设置 RMDH1S。

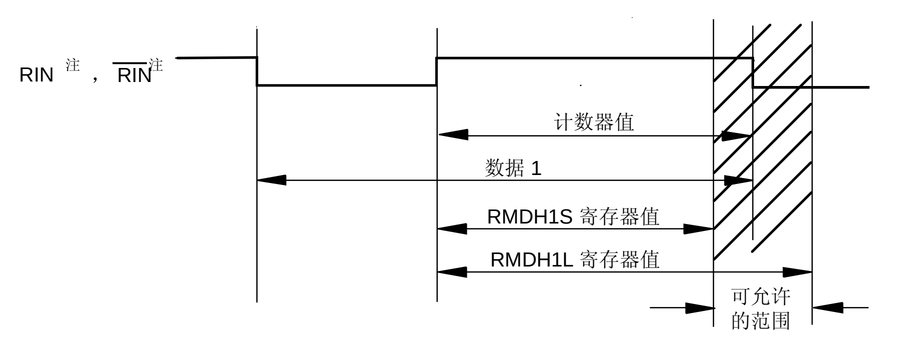
复位信号的发生会将 RMDH1S 设置为 00H。

（13） 遥控器接收 DH1L 比较寄存器（RMDH1L）

寄存器用于检测遥控器数据 1（长侧）的高电平。

可以通过 8 位存储器操作指令来设置 RMDH1L。

复位信号的发生会将 RMDH1L 设置为 00H。



注　在类型 A 接收模式下产生 RIN，在类型 B 和类型 C 接收模式下产生 $\overline{\text{RIN}}$。

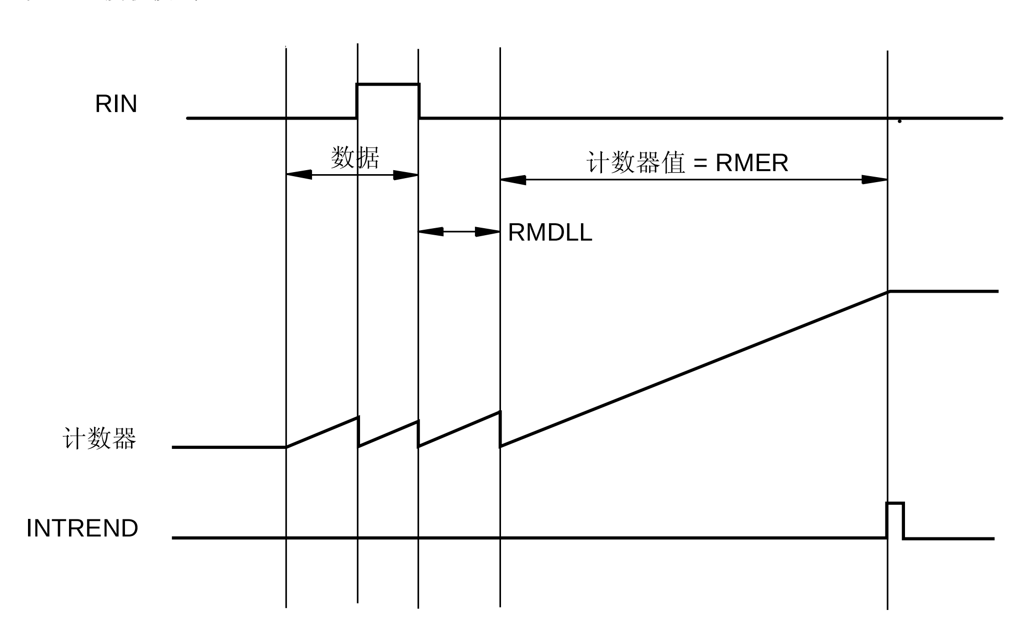
（14）遥控器接收结束宽度选择寄存器（RMER）

该寄存器决定输出 INTREND 信号的时序间隔。

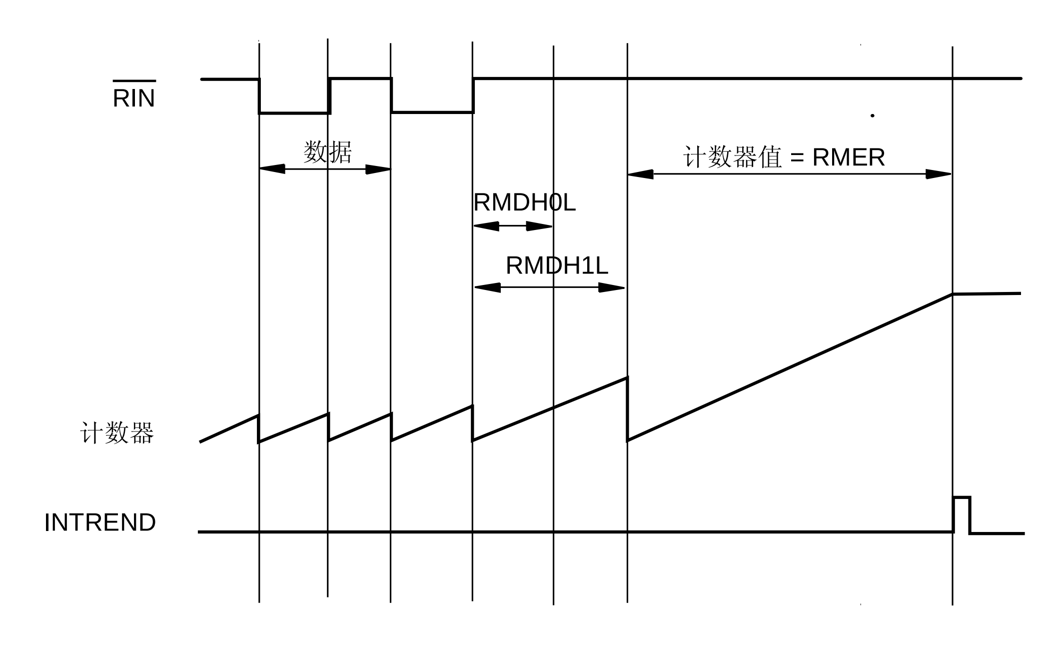
可以通过 8 位存储器操作指令来设置 RMER。

复位信号的发生会将 RMER 设置为 00H。

（a）类型 A 接收模式



（b）类型 B，类型 C 接收模式



注意事项　对于 **RMER** 和所有的遥控器接收比较寄存器（**RMGPLS**，**RMGPLL**，**RMGPHS**，**RMGPHL**，**RMDLS**，**RMDLL**，**RMDH0S**，**RMDH0L**，**RMDH1S** 和 **RMDH1L**），首先禁止遥控器接收（遥控器接收控制寄存器（**RMCN**）的第 **7** 位（**RMEN**）**= 0**），然后改变该值。

19.3 控制遥控器接收器的寄存器

遥控器接收器由下列寄存器控制。

- 遥控器接收中断状态寄存器（INTS）
- 遥控器接收中断状态清除寄存器（INTC）
- 遥控器接收控制寄存器（RMCN）

（1）遥控器接收中断状态寄存器（INTS）

该寄存器用于在遥控接收中断（INTRERR，INTGP，INTREND，INTDFULL）中识别已经发生的中断请求。
可以通过 1 位或 8 位存储器操作指令来设置 INTS。
复位信号的发生会将 INTS 设置为 00H。

图 19-3. 遥控器接收中断状态寄存器（INTS）的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
INTS	0	0	0	0	INTS DFULL	INTS REND	INTS GP	INTS RERR	FFF9H	00H	R

INTS DFULL	因为读取 8 位移位数据的中断请求
0	未发生因读取 8 位移位数据的中断请求
1	已发生因读取 8 位移位数据的中断请求

INTS REND	数据接收完成中断请求
0	未发生数据接收完成中断请求
1	已发生数据接收完成中断请求

INTS GP	引导脉冲检测中断
0	未发生引导脉冲检测中断
1	已发生引导脉冲检测中断

INTS RERR	遥控接收错误中断请求
0	未发生遥控接收错误中断请求
1	已发生遥控接收错误中断请求

注意事项　即使读取 INTS 寄存器，它也不会被清除。使用 INTC 寄存器来清除 INTS 寄存器。

（2）遥控器接收中断状态清除寄存器（INTC）

该寄存器用于控制遥控器接收中断状态寄存器（INTS）。

可以通过 1 位或 8 位存储器操作指令来设置 INTC。

复位信号的发生会将 INTC 设置为 00H。

图 19-4. 遥控器接收中断状态清除寄存器（INTC）的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
INTC	0	0	0	0	INTC DFULL	INTC REND	INTC GP	INTC RERR	FFFAH	00H	R/W

INTC DFULL	由读取 8 位移位数据控制的中断指示位
0	INTSDFULL 位未改变
1	INTSDFULL 位被清除

INTC REND	数据接收完成中断指示位的控制
0	INTSREND 位未改变
1	INTSREND 位被清除

INTC GP	引导脉冲检测中断指示位的控制
0	INTSGP 位未改变
1	INTSGP 位被清除

INTC RERR	由遥控接收错误控制的中断指示位
0	INTSRERR 位未改变
1	INTSRERR 位被清除

（**3**）遥控器接收控制寄存器（**RMCN**）

该寄存器用于使能/禁止遥控器接收，并设置噪声消除宽度，设置时钟内部分频，设置输入反转信号和源时钟。
可以通过 1 位或 8 位存储器操作指令来设置 RMCN。
复位信号的发生会将 RMCN 设置为 00H。

图 19-5. 遥控器接收控制寄存器（RMCN）的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
RMCN	RMEN	NCW	PRSEN	RMIN	RMMD1	RMMD0	RMCK1	RMCK0	FF9AH	00H	R/W

RMEN	遥控器接收操作的控制
0	禁止遥控器接收
1	使能遥控器接收

NCW	噪声消除宽度控制信号
0	消除噪声小于 $1/f_{REMPRS}$
1	消除噪声小于 $2/f_{REMPRS}$

PRSEN	内部时钟分频控制信号
0	无内部分频（$f_{REMPRS} = f_{REM}$）
1	内部 2 分频（$f_{REMPRS} = f_{REM}/2$）

RMIN	遥控器输入反转信号
0	输入正相位
1	输入反相位

RMMD1	RMMD0	遥控器接收模式
0	0	类型 A 接收模式（提供引导脉冲（半个时钟））
0	1	类型 B 接收模式（提供引导脉冲（一个时钟））
1	0	类型 C 接收模式（不提供引导脉冲）
1	1	禁止设置

RMCK1	RMCK0	选择遥控器计数器的源时钟（f_{REM}）的选择
0	0	$f_{PRS}/2^6$（156.25 kHz）
0	1	$f_{PRS}/2^7$（78.125 kHz）
1	0	$f_{PRS}/2^8$（39.063 kHz）
1	1	f_{SUB}（32.768 kHz）

注意事项　**要改变 NCW, PRSEN, RMIN, RMMD1, RMMD0, RMCK1 和 RMCK0 的值，首先要禁止遥控器接收（RMEN = 0）。**

备注　**1.** f_{REM}：遥控器计数器的源时钟（由第 0 位和第 1 位（RMCK0 和 RMCK1）选择）

备注　**2.** f_{REMPRS}：　遥控器接收器内的操作时钟
3. f_{PRS}：　外设硬件时钟频率
4. f_{SUB}：　副系统时钟的振荡频率
5. 括号中的值适用于 f_{PRS} = 10 MHz 且 f_{SUB} = 32.768 kHz。

19.4 遥控器接收器的操作

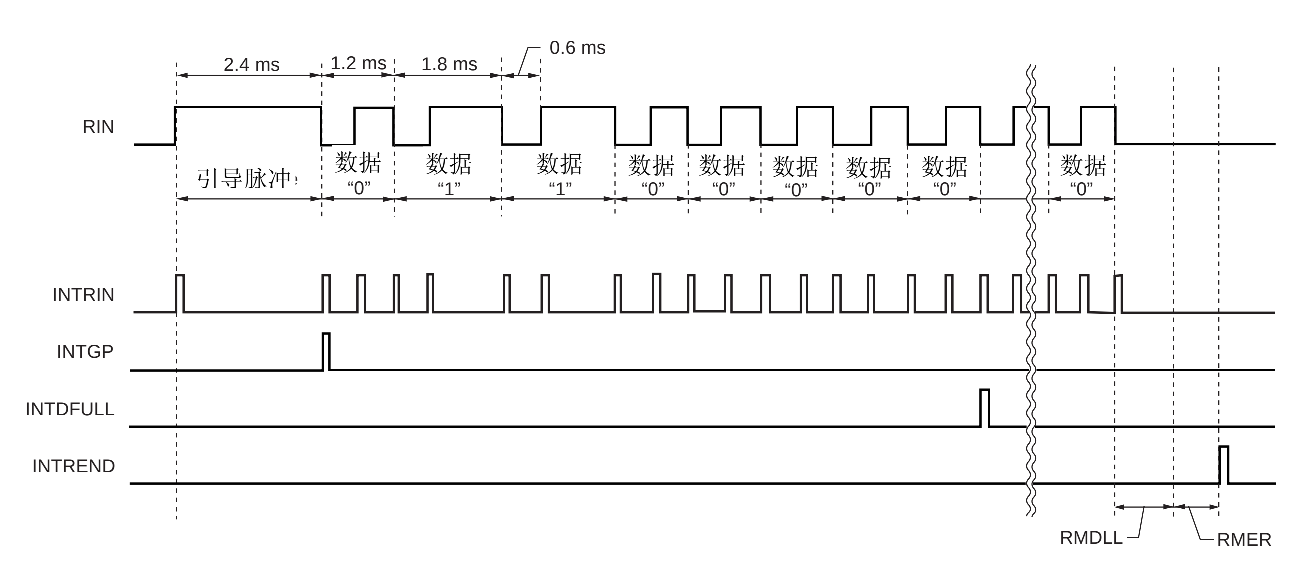
该遥控器接收器可以使用下列遥控器接收模式。

- 具有引导脉冲（半个时钟）的类型 A 接收模式
- 具有引导脉冲（一个时钟）的类型 B 接收模式
- 不具有引导脉冲的类型 C 接收模式

19.4.1 类型A接收模式的格式

图 19-6 展示了类型 A 的数据格式。

图 19-6. 类型 A 数据格式示例



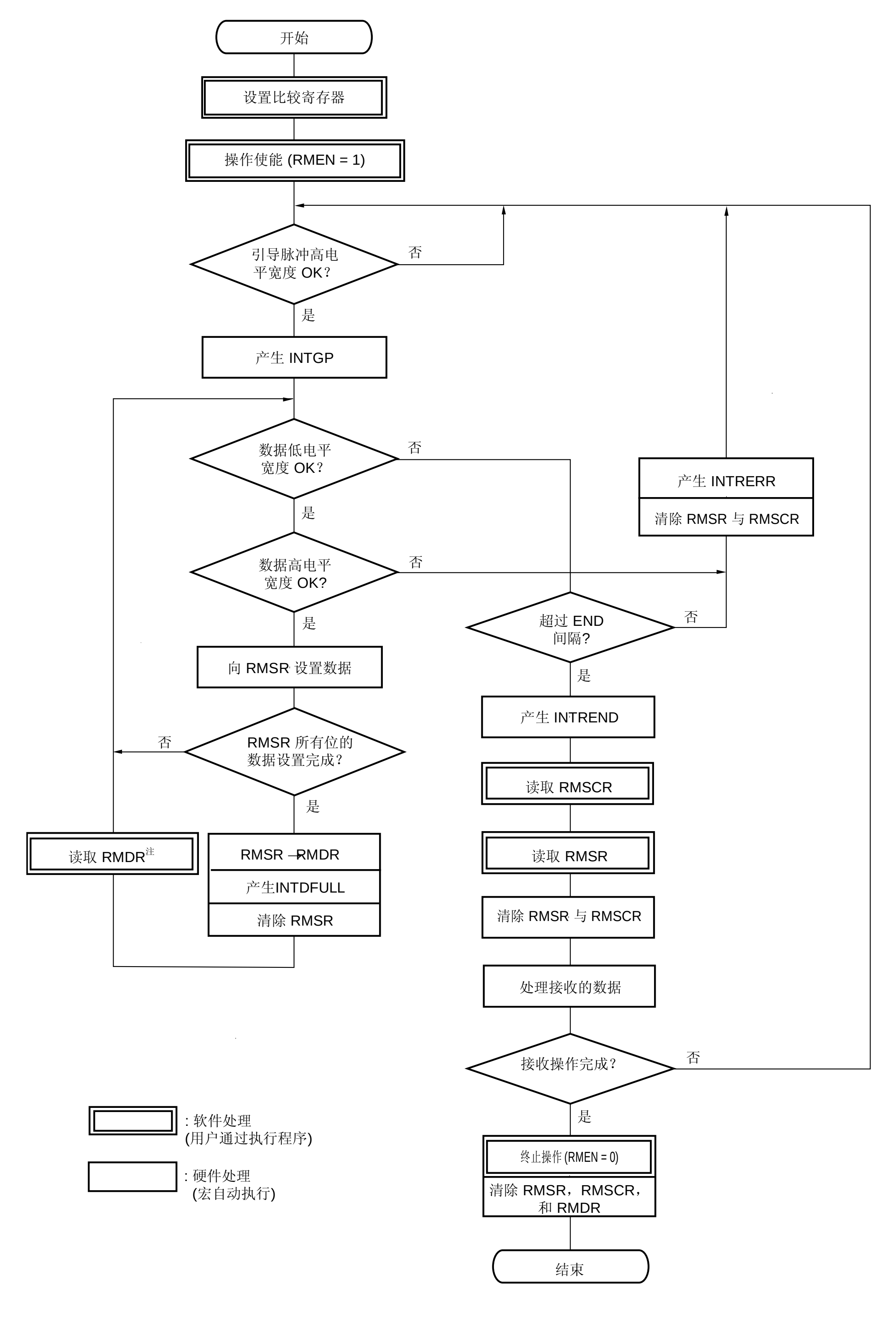
19.4.2 类型 A接收模式的操作流程

图 19-7 展示了操作流程。

注意事项 **1.** 产生 **INTRERR** 时，**RMSR** 和 **RMSCR** 立即被自动清除。
2. 当数据已经设置到 **RMSR** 的所有位时，自动执行如下处理。
- **RMSR** 的值被传送到 **RMDR**。
- 产生 **INTDFULL**。
- **RMSR** 被清除。

在下一次数据设置到 **RMSR** 的所有位之前，必须读取 **RMDR**。
3. 当 **INTREND** 产生后，在 **RMSR** 之后首先读取 **RMSCR**。
读取 **RMSR** 后，**RMSCR** 和 **RMSR** 被自动清除。
如果产生 **INTREND**，不能接收下一个数据，直到 **RMSR** 被读取。
4. **RMSR**，**RMSCR** 和 **RMDR** 被同时清除，可以结束操作（**RMEN = 0**）。

图 19-7. 类型 A 接收模式的操作流程



注　　在数据设置到 RMSR 的所有位之前，必须读取 RMDR。

19.4.3　类型B接收模式的格式

图 19-8 展示了类型 B 的数据格式。

图 19-8. 类型 **B** 数据格式示例

备注　　$\overline{\text{RIN}}$ 是 RIN 的内部反转信号。

19.4.4 类型B接收模式的操作流程

图 19-9 展示了操作流程。

注意事项 **1.** 产生 **INTRERR** 时，**RMSR** 和 **RMSCR** 立即被自动清除。

2. 当数据已经设置到 **RMSR** 的所有位时，自动执行如下处理。

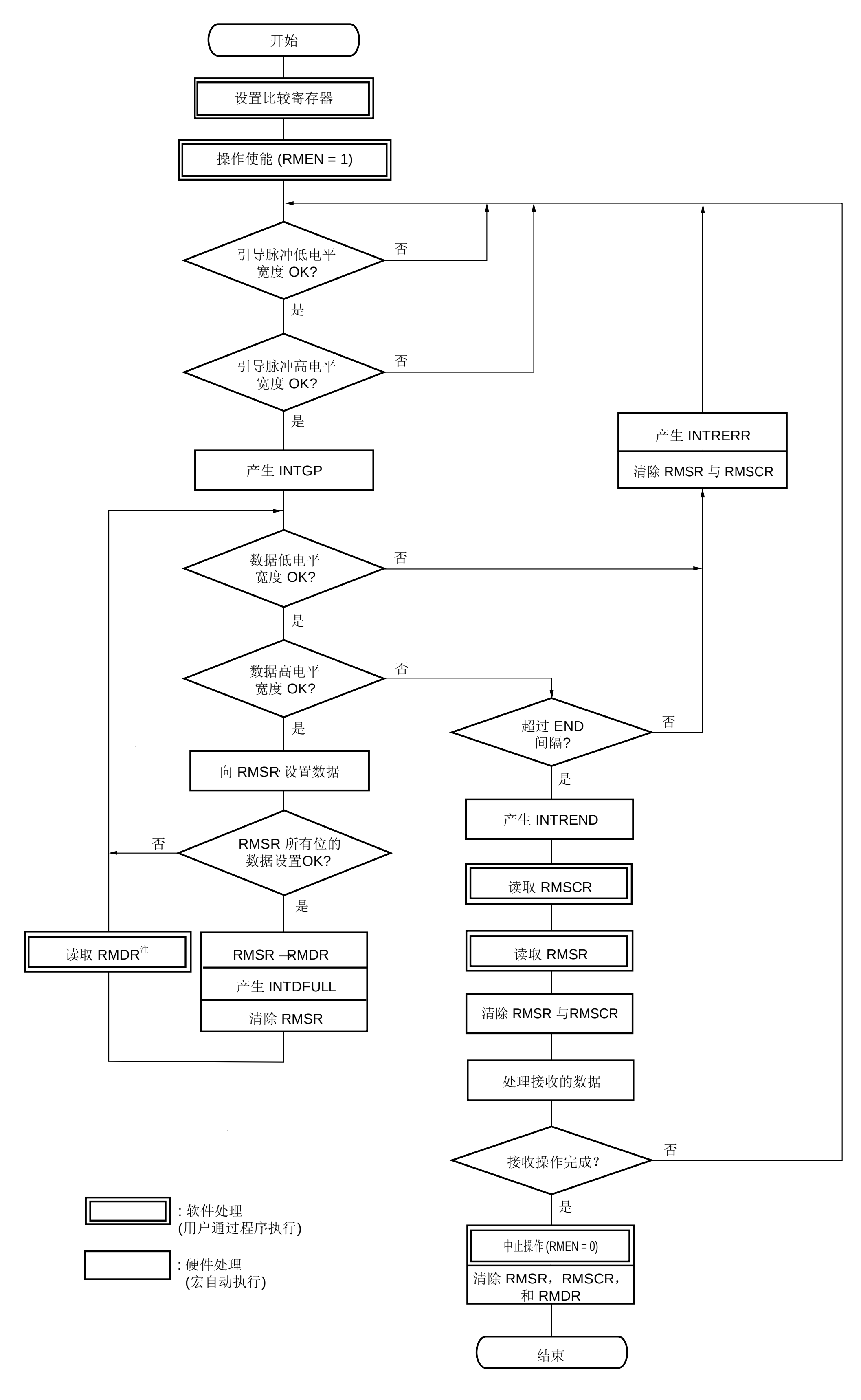
- **RMSR** 的值被传送到 **RMDR**。
- 产生 **INTDFULL**。
- **RMSR** 被清除。

在下一次数据设置到 **RMSR** 的所有位之前，必须读取 **RMDR**。

3. 当 **INTREND** 产生后，在 **RMSR** 之后首先读取 **RMSCR**。
读取 **RMSR** 后，**RMSCR** 和 **RMSR** 被自动清除。
如果产生 **INTREND**，不能接收下一个数据，直到 **RMSR** 被读取。

4. **RMSR**，**RMSCR** 和 **RMDR** 被同时清除，可以结束操作（**RMEN = 0**）。

图 19-9. 类型 B 接收模式的操作流程



注　　在数据设置到 RMSR 的所有位之前，必须读取 RMDR。

19.4.5 类型C接收模式的格式

图 19-10 展示了类型 C 数据格式 。

图 19-10. 类型 C 数据格式的示例

0.263 ms
1.0 ms　2.0 ms
RIN
$\overline{\text{RIN}}$
数据 "0"　数据 "1"　数据 "1"　数据 "0"　数据 "0"　数据 "0"　数据 "0"　数据 "0"　数据 "0"
INTRIN
INTGP
INTDFULL
INTREND
RMDH1L
RMER

备注　　$\overline{\text{RIN}}$ 是 RIN 的内部反转信号。

19.4.6 类型C接收模式的操作流程

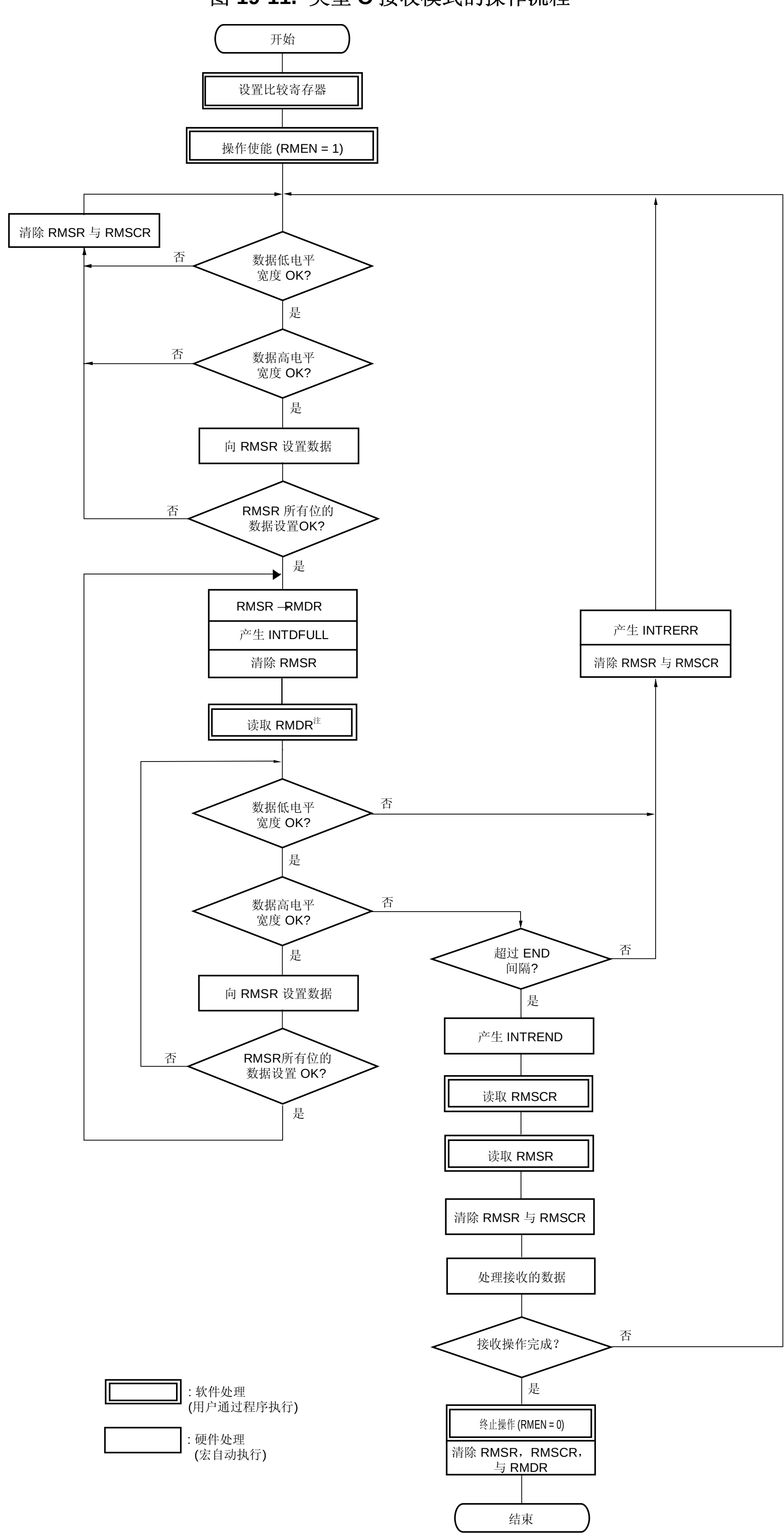
图 19-11 展示了操作流程。

注意事项

1. 产生 **INTRERR** 时，**RMSR** 和 **RMSCR** 立即被自动清除。
2. 当数据已经设置到 **RMSR** 的所有位时，自动执行如下处理。
   - **RMSR** 的值被传送到 **RMDR**。
   - 产生 **INTDFULL**。
   - **RMSR** 被清除。

   在下一次数据设置到 **RMSR** 的所有位之前，必须读取 **RMDR**。
3. 当 **INTREND** 产生后，在 **RMSR** 之后首先读取 **RMSCR**。
   读取 **RMSR** 后，**RMSCR** 和 **RMSR** 被自动清除。
   如果产生 **INTREND**，不能接收下一个数据，直到 **RMSR** 被读取。
4. **RMSR**，**RMSCR** 和 **RMDR** 被同时清除，可以结束操作（**RMEN = 0**）。
5. 在类型 **C** 接收模式下，如果在第一个 **INTDFULL** 中断产生之前，不满足接收数据低**/**高电平宽度的条件，将不会产生 **INTRERR** 和 **INTREND**。但是，**RMSR** 和 **RMSCR** 将会被清除。

图 19-11. 类型 C 接收模式的操作流程

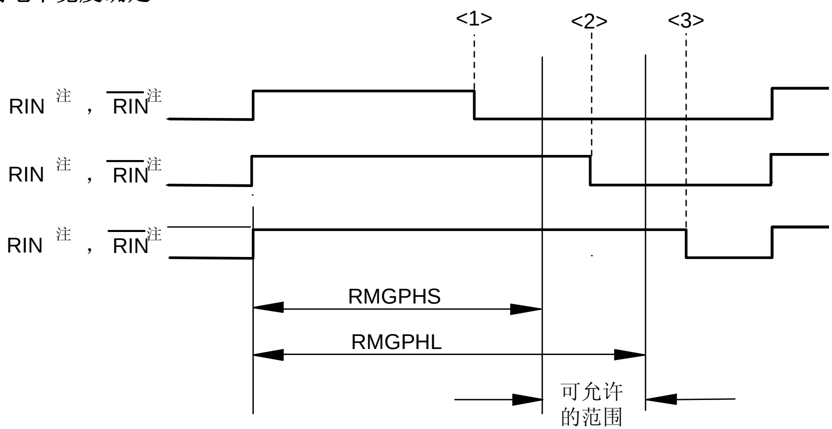


注　　在数据设置到 RMSR 的所有位之前，必须读取 RMDR。

19.4.7 时序

操作的变化取决于下列 RIN 输入波形的位置。（仅限类型 A，类型 B 接收模式）

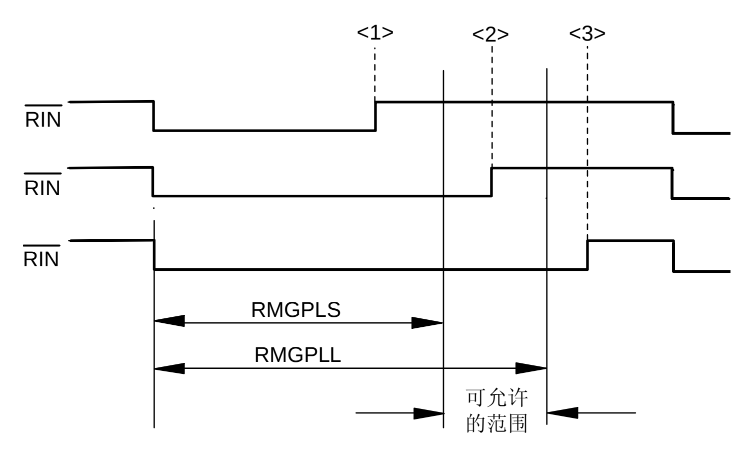
（1）引导脉冲高电平宽度确定



注　　类型 A 接收模式下产生 RIN，类型 B 接收模式下产生 $\overline{\text{RIN}}$。

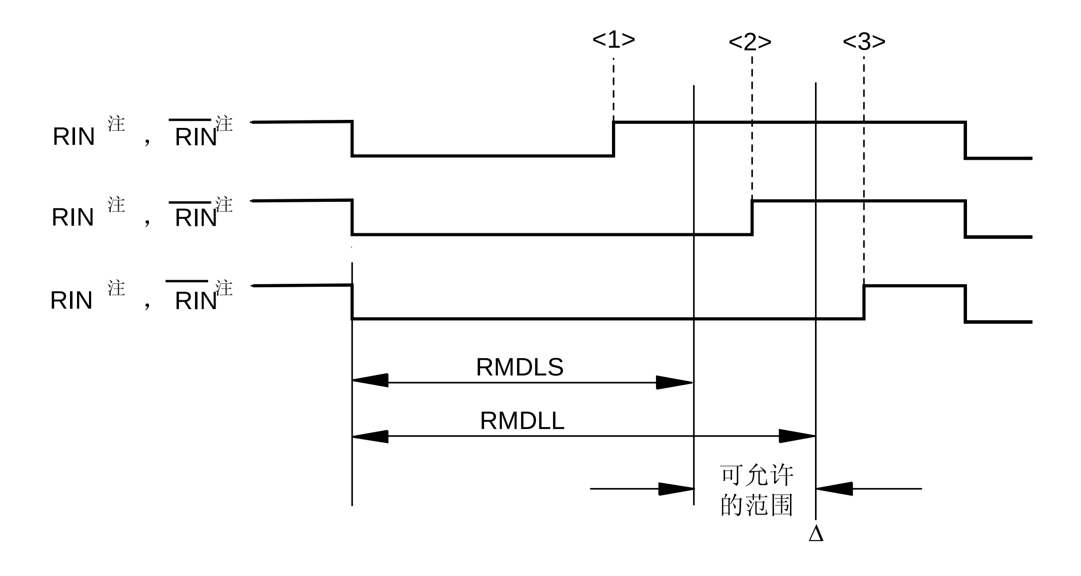
RMGPHS/RMGPHL/计数器之间的关系	波形位置	相关操作
计数器 < RMGPHS	<1>：短	从下一个上升沿开始测量引导脉冲高电平宽度。
RMGPHS ≤ 计数器< RMGPHL	<2>：范围内	产生 INTGP。 数据测量开始。
RMGPHL ≤ 计数器	<3>：长	从下一个上升沿开始测量引导脉冲高电平宽度。

（2）引导脉冲高电平宽度确定（仅限类型 B 接收模式）



RMGPLS/RMGPLL/计数器之间的关系	波形位置	相关操作
计数器 < RMGPLS	<1>：短	从下一个上升沿开始测量引导脉冲高电平宽度。
RMGPLS ≤ 计数器< RMGPLL	<2>：范围内	产生 INTGP。 数据测量开始。
RMGPLL ≤ 计数器	<3>：长	从下一个上升沿开始测量引导脉冲高电平宽度。

（3）数据低电平宽度确定

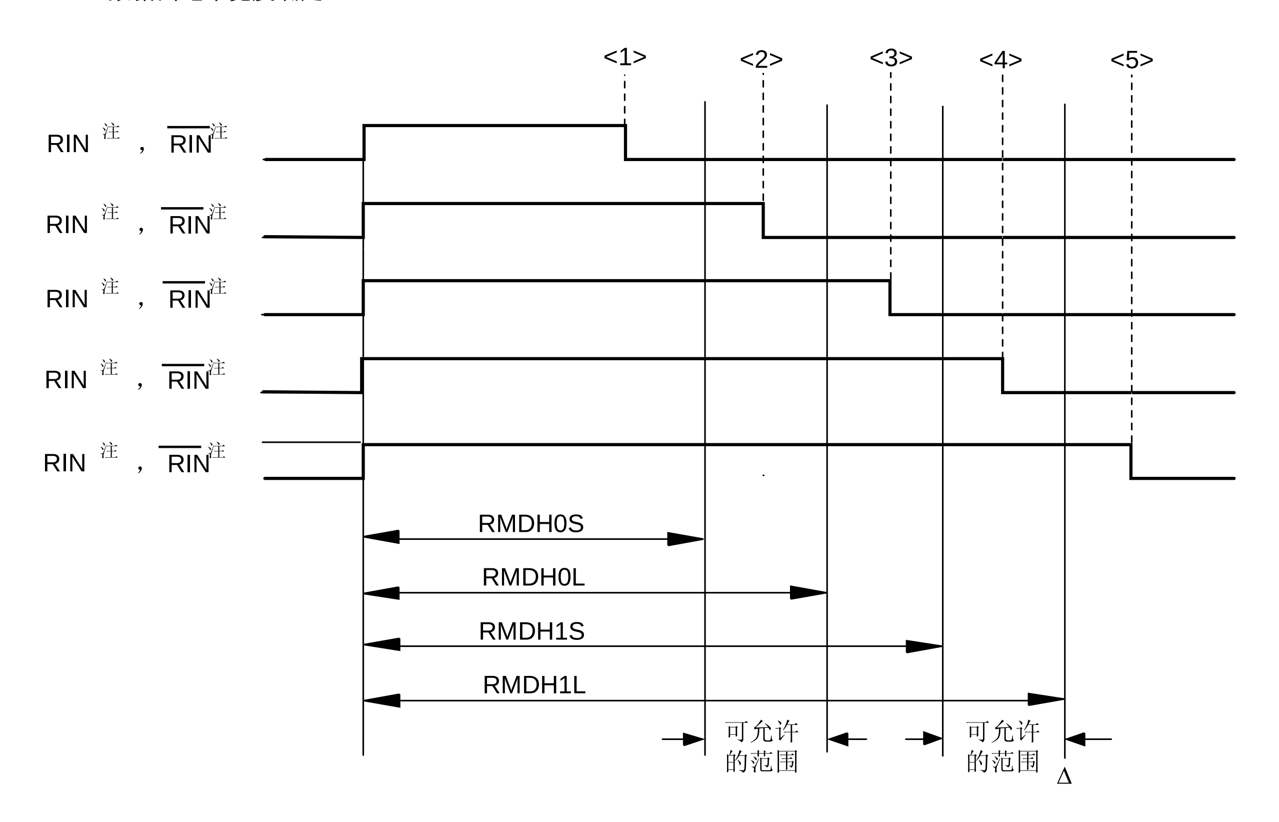


注　　类型 A 接收模式下产生 RIN，类型 B 和类型 C 接收模式下产生 $\overline{\text{RIN}}$。

RMDLS/RMDLL/计数器之间的关系	波形位置	相关操作
计数器 < RMDLS	<1>：短	产生错误中断 INTRERR 注。 开始测量引导脉冲高电平宽度。
RMDLS ≤ 计数器< RMDLL	<2>：范围内	开始测量数据高电平宽度。
RMDLL ≤ 计数器	<3>：长	（类型 A 接收模式） 从 Δ 点开始测量结束宽度 （类型 B，类型 C 接收模式） 在 Δ 点产生产生错误中断 INTRERR 注

注　　类型 C 接收模式下，在产生第一个 INTDFULL 中断之前，不产生 INTRERR。但是，RMSR 和 RMSCR 将会被清除。

（**4**）数据高电平宽度确定



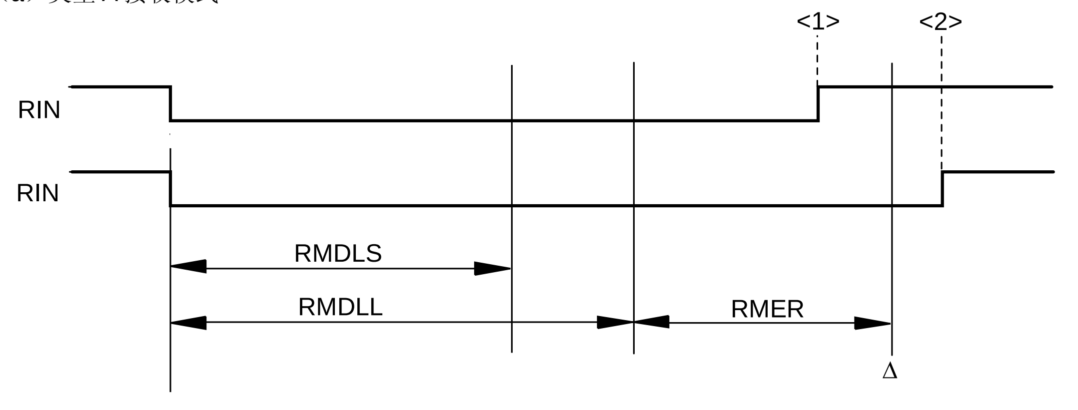
注 类型 A 接收模式下产生 RIN，类型 B 和类型 C 接收模式下产生 $\overline{RIN}$。

RMDH0S/RMDH0L/RMDH1S/RMDH1L/计数器之间的关系	波形的位置	相应操作
计数器 < RMDH0S	<1>： 短	产生错误中断 INTRERR。 从下一个上升沿开始测量引导脉冲高电平宽度。
RMDH0S ≤计数器< RMDH0L	<2>：范围内	接收到数据 0。 开始测量数据低电平宽度。
RMDH0L ≤计数器< RMDH1S	<3>： 超出范围	产生错误中断 INTRERR。 从下一个上升沿开始测量引导脉冲高电平宽度。
RMDH1S ≤计数器< RMDH1L	<4>： 范围内	接收到数据 1。 开始测量数据低电平宽度。
RMDH1L ≤计数器	<5>： 长	（类型 A 接收模式） 在 Δ 点产生产生错误中断 INTRERR 注 （类型 B，类型 C 接收模式） 从 Δ 点开始测量结束宽度。从下一个上升沿开始测量引导脉冲高电平宽度。

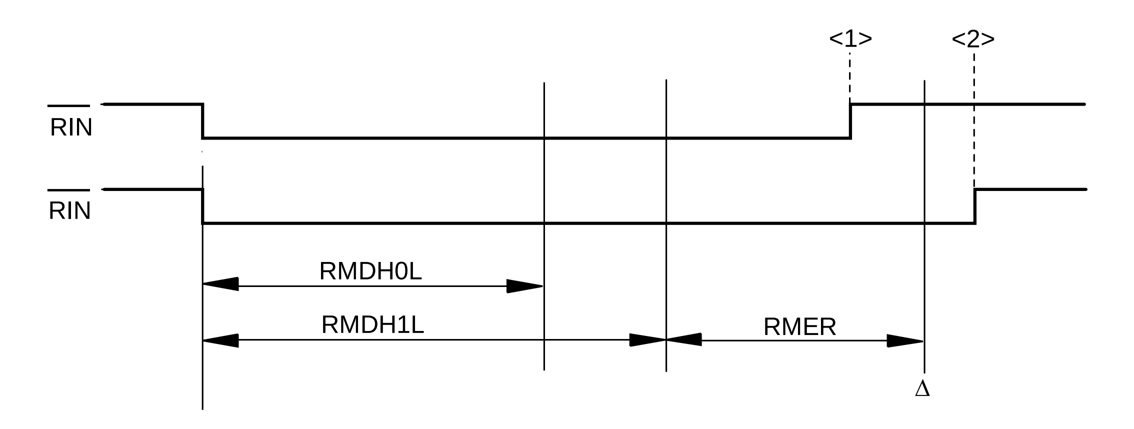
注 类型 C 接收模式下，在产生第一个 INTDFULL 中断之前，不产生 INTRERR。但是，RMSR 和 RMSCR 将会被清除。

（5）结束宽度确定

（a）类型 A 接收模式



（b）类型 B，类型 C 接收模式



RMER/计数器之间的关系	波形位置	对应的操作
计数器< RMER	<1>：短	产生错误中断 INTRERR。 开始测量引导脉冲高电平宽度。
RMER ≤ 计数器	<2>：长	在 Δ 点[注]产生 INTREND。 通过电路的接收停止，直到 RMSR 被读取。

注　类型 C 接收模式下，在产生第一个 INTDFULL 中断之前，不产生 INTRERR 和 INTREND。但是，RMSR 和 RMSCR 将会被清除。

19.4.8 比较寄存器设定

遥控器接收器有如下 11 种类型比较寄存器。

- 遥控器接收 GPLS 比较寄存器（RMGPLS）
- 遥控器接收 GPLL 比较寄存器（RMGPLL）
- 遥控器接收 GPHS 比较寄存器（RMGPHS）
- 遥控器接收 GPHL 比较寄存器（RMGPHL）
- 遥控器接收 DLS 比较寄存器（RMDLS）
- 遥控器接收 DLL 比较寄存器（RMDLL）
- 遥控器接收 DH0S 比较寄存器（RMDH0S）
- 遥控器接收 DH0L 比较寄存器（RMDH0L）
- 遥控器接收 DH1S 比较寄存器（RMDH1S）
- 遥控器接收 DH1L 比较寄存器（RMDH1L）
- 遥控器接收结束宽度选择寄存器（RMER）

使用下面的公式（1）到（3）来设置各个比较寄存器的值。

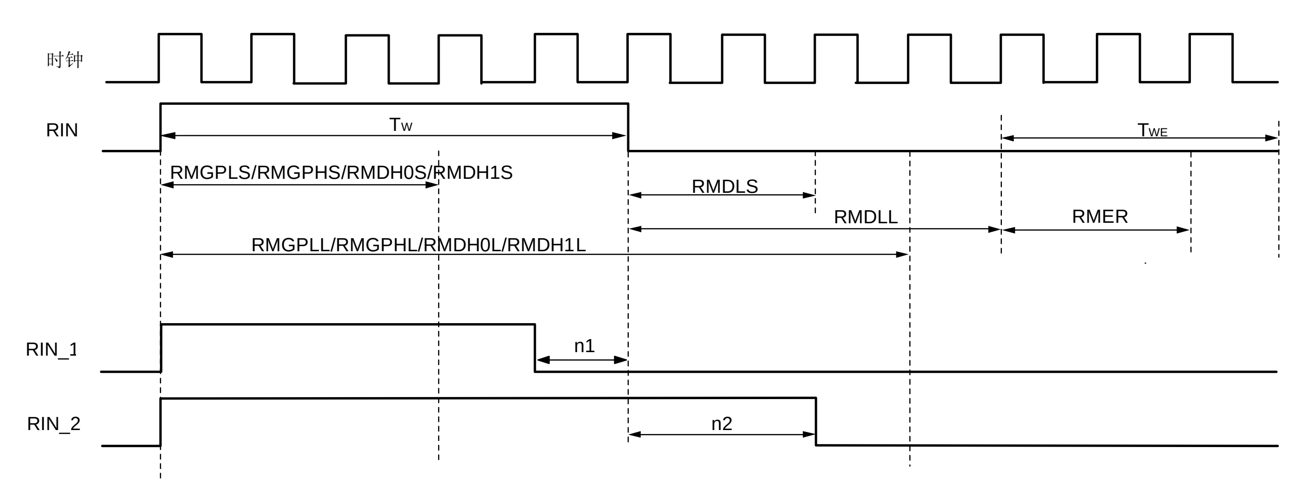
要留出容差的容限，即使因为噪声的影响，RIN 的输入波形如图 19-12 所示的 RIN_1 或 RIN_2，也能确保正常的接收操作。

注意事项 **1.** 总是在遥控器接收被禁止时（**RMEN = 0**）设置各个比较寄存器。

2. 设置的值要满足下列四个条件。

- **RMGPLS < RMGPLL**
- **RMGPHS < RMGPHL**
- **RMDLS < RMDLL**
- **RMDH0S < RMDH0L ≤ RMDH1S < RMDH1L**

图 19-12. 设置示例（当 n1 = 1，n2 = 2）



（1）RMGPLS，RMGPHS，RMDLS，RMDH0S 和 RMDH1S 的公式

$$\left[\frac{T_W \times (1 - a/100)}{1/f_{REMPRS}}\right]_{INT} - 2 - n1$$

（2）RMGPLL， RMGPHL，RMDLL，RMDH0L 和 RMDH1L 的公式

$$\left[\frac{T_W \times (1 + a/100)}{1/f_{REMPRS}}\right]_{INT} + 1 + n2$$

（3）RMER 的公式

$$\left[\frac{T_{WE} \times (1 - a/100)}{1/f_{REMPRS}}\right]_{INT} - 1$$

T_W：　RIN 输入波形的宽度

$1/f_{REMPRS}$：　在 PRSEN 控制的分频后，内部操作时钟周期的宽度

a：　容差（%）

[ ] INT：　将方括号中公式计算的值舍掉小数部分。

n1，n2：　由噪声引起的波形变化注 1

T_{WE}：　RIN 输入的结束宽度注 2

注　**1.** 按照用户系统规格的需求来设置 n1 和 n2 的值。

2. 在 RMDLL 之后计算的结束宽度。

在最后的数据已经接收后实际需要的低电平宽度，如下所示：

（RMDLL + 1 + RMER + 1）×（由 PRSEN 控制的分频后，内部操作时钟周期的宽度）

19.4.9 错误中断产生时序

（1）类型 A 接收模式

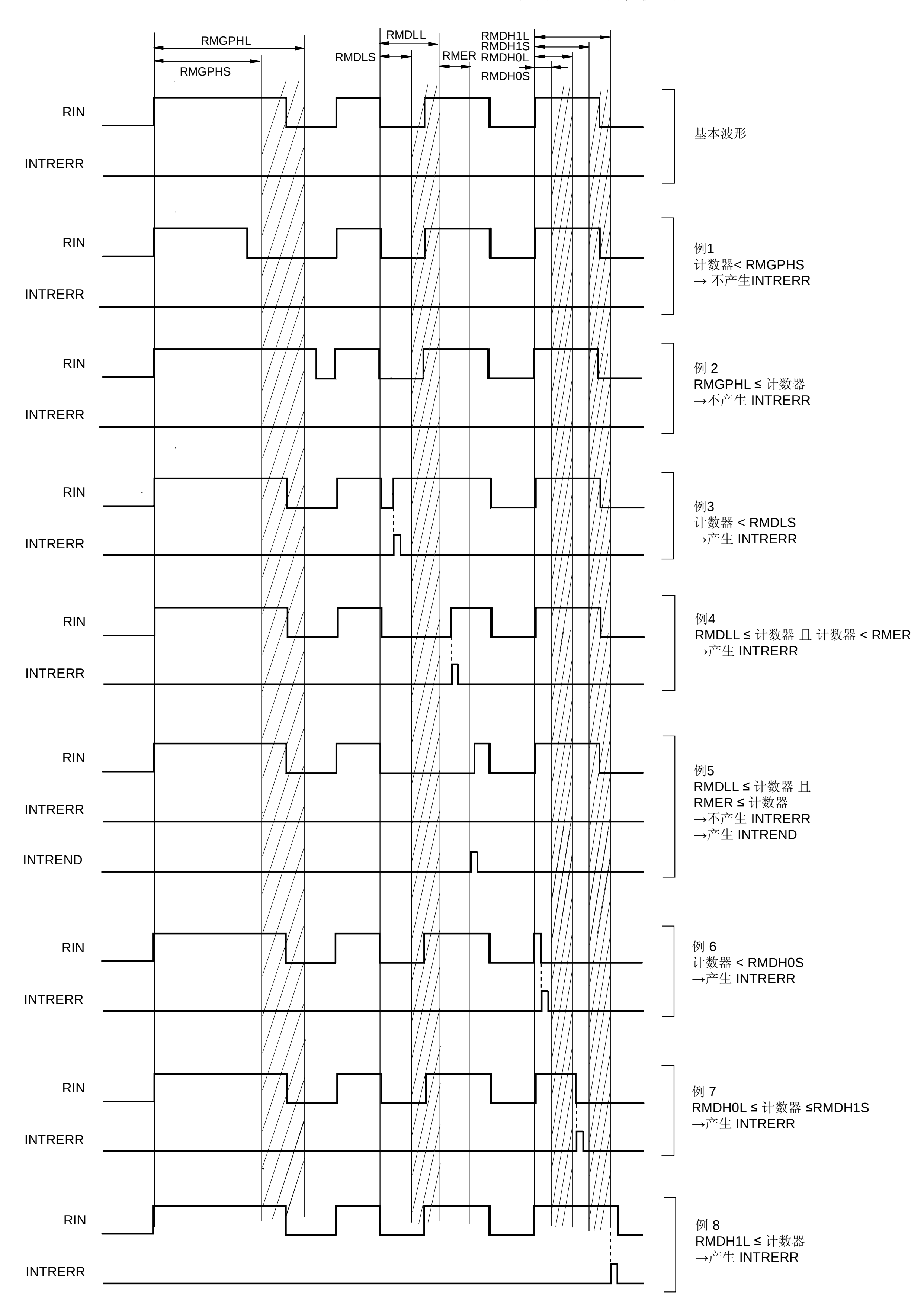
在正常检测到引导脉冲后，在下列任何条件下，都会产生 INTRERR 信号。

- 计数器 < RIN 上升沿时的 RMDLS
- RMDLL ≤ 计数器且 RMDLL 之后的计数器< RIN 上升沿时的 RMER
- 计数器 < RIN 下降沿时的 RMDH0S
- RMDH0L ≤ 计数器 < RIN 下降沿时的 RMDH1S
- 当 RIN 为高电平时，寄存器导致 RMDH1L ≤ 计数器

不产生 INTRERR 信号，直到检测到引导脉冲。

一旦产生了 INTRERR 信号，它将不会再次产生，直到检测到下一个引导脉冲。INTRERR 信号的产生时序如图 19-13 所示。

图 **19-13. INTRERR** 信号的产生时序（类型 **A** 接收模式）



（2）类型 B 接收模式

在正常检测到引导脉冲后，在下列任何条件下，都会产生 INTRERR 信号。

- 计数器 < $\overline{\text{RIN}}$ 上升沿时的 RMDLS
- 当 $\overline{\text{RIN}}$ 为低电平时，寄存器导致 RMDLL ≤ 计数器
- 计数器 < $\overline{\text{RIN}}$ 下降沿时的 RMDH0S
- RMDH0L ≤ 计数器 < $\overline{\text{RIN}}$ 下降沿时的 RMDH1S
- RMDH1L ≤ 计数器且 RMDH1L 之后的计数器< $\overline{\text{RIN}}$ 下降沿时的 RMER

不产生 INTRERR 信号，直到检测到引导脉冲。

一旦产生了 INTRERR 信号，它将不会再次产生，直到检测到下一个引导脉冲。INTRERR 信号的产生时序如图 19-14 所示。

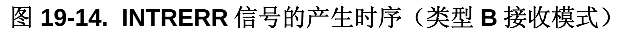


图 **19-14. INTRERR** 信号的产生时序（类型 **B** 接收模式）

RMGPHL
RMDLL
RMDH1L
RMDH1S
RMDH0L
RMDH0S
RMGPLL
RMGPHS
RMDLS
RMER
RMGPLS

RIN
INTRERR
基本波形

RIN
INTRERR
例1
计数器 < RMGPLS
→不产生 INTRERR

RIN
INTRERR
例 2
RMGPLL≤ 计数器
→不产生 INTRERR

RIN
INTRERR
例3
计数器 < RMGPHS
→不产生 INTRERR

RIN
INTRERR
例 4
RMGPHL ≤ 计数器
→不产生 INTRERR

RIN
INTRERR
例 5
计数器 < RMDLS
→产生 INTRERR

RIN
INTRERR
例 6
RMDLL ≤ 计数器
→产生 INTRERR

RIN
INTRERR
例 7
计数器 < RMDH0S
→产生 INTRERR

RIN
INTRERR
例 8
RMDH0L ≤ 计数器 < RMDH1S
→产生 INTRERR

RIN
INTRERR
例 9
RMDH1L ≤ 计数器
且 计数器 < RMER
→产生 INTRERR

RIN
INTRERR
INTREND
例 10
RMDLL ≤ 计数器
且 RMER ≤ 计数器
→不产生 INTRERR
→产生 INTREND

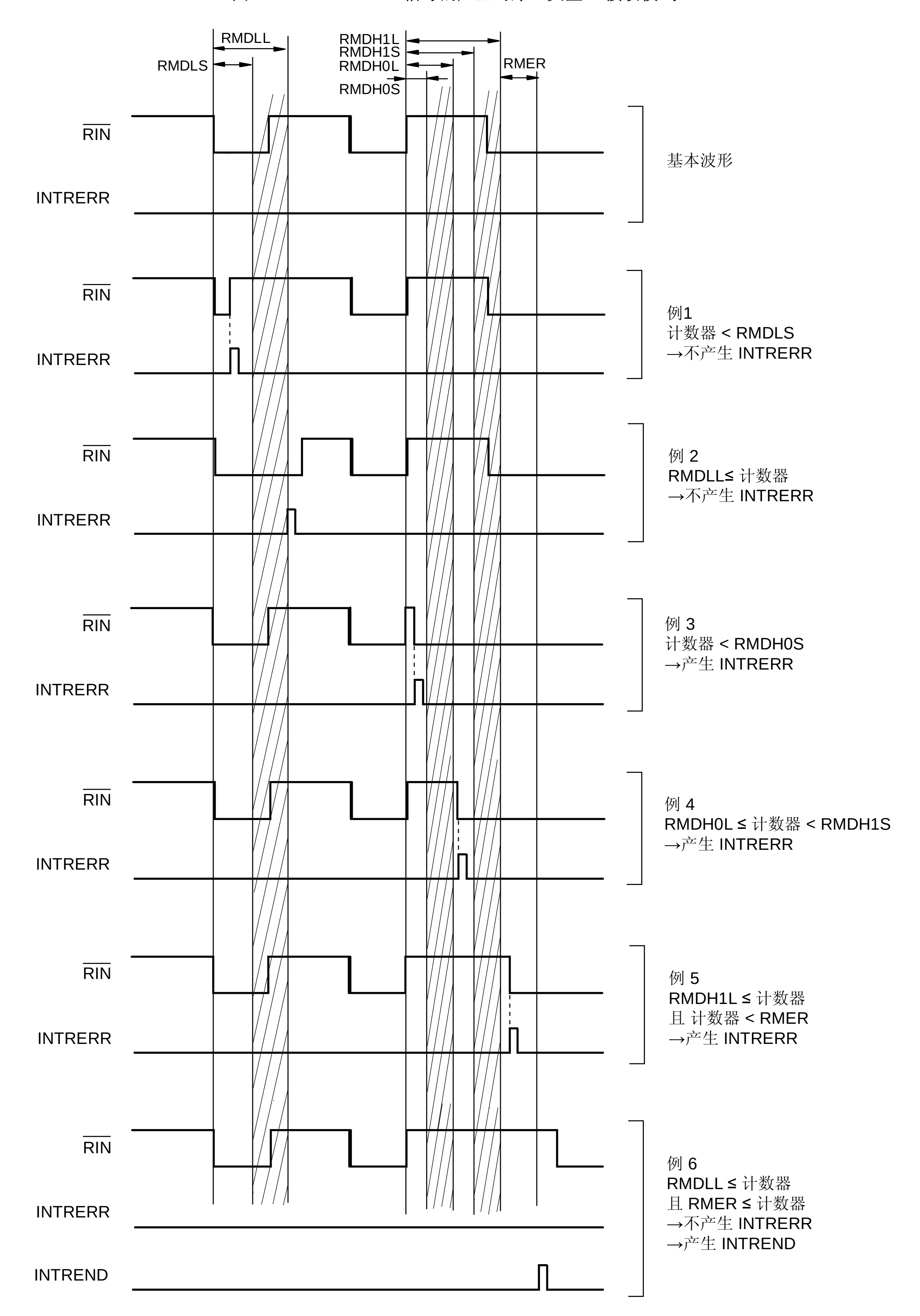
(3)类型 C 接收模式

在下列任何条件下，都会产生 INTRERR 信号。

- 计数器 < $\overline{\text{RIN}}$ 上升沿时的 RMDLS
- 当 $\overline{\text{RIN}}$ 为低电平时，寄存器导致 RMDLL ≤ 计数器
- 计数器 < $\overline{\text{RIN}}$ 下降沿时的 RMDH0S
- RMDH0L ≤ 计数器 < $\overline{\text{RIN}}$ 下降沿时的 RMDH1S
- RMDH1L ≤ 计数器且 RMDH1L 之后的计数器< $\overline{\text{RIN}}$ 下降沿时的 RMER

但是，在产生第一个 INTDFULL 中断之前，不产生 INTRERR 信号。
INTRERR 信号的产生时序如图 19-15 所示。

图 **19-15. INTRERR** 信号的产生时序（类型 **C** 接收模式）



19.4.10 噪声消除

遥控器接收器提供噪声消除功能，对来自外部的输入信号消除噪声后，再提供给 RIN 引脚。

通过设置遥控接收控制寄存器（RMCN）的第 5 位（PRSEN）和第 6 位（NCW），可以选择消除噪声的宽度，如表 19-2 所示。

表 19-2. 噪声消除宽度

PRSEN 分频控制信号	NCW 噪声消除宽度信号	在 PRSEN 控制的分频后，内部操作时钟周期（$1/f_{REMPRS}$）	消除噪声宽度
0	0	$1/f_{REM}$	小于 $1/f_{REM}$
0	1	$1/f_{REM}$	小于 $2/f_{REM}$
1	0	$2/f_{REM}$	小于 $2/f_{REM}$
1	1	$2/f_{REM}$	小于 $4/f_{REM}$

备注　f_{REM}：遥控器计数器的源时钟

通过使用 PRSEN 控制分频后的内部操作时钟来执行噪声消除操作。

然后，从 RIN 引脚输入的外部信号与时钟同步后，

如果 NCW = 0，在电路中执行两次采样后的信号被处理作为 RIN 的输入。

如果 NCW = 1，在电路中执行三次采样后的信号被处理作为 RIN 的输入。

噪声消除操作的流程如下所示。

<1>　选择内部操作时钟是否通过 PRSEN 分频。
　PRSEN = 0：　不分频（$f_{REMPRS} = f_{REM}$）
　PRSEN = 1：　分频（$f_{REMPRS} = f_{REM}/2$）

<2>　将由 RIN 引脚输入的外部信号与内部操作时钟同步。

<3>　产生信号（samp1）对已经同步的信号进行第一次采样。
　（该信号比已经同步的信号晚一个时钟。）

<4>　产生信号（samp2）对已经同步的信号和 samp1 进行第二次采样。
　（当已经同步的信号 = samp1 = H，samp1 被锁存。）

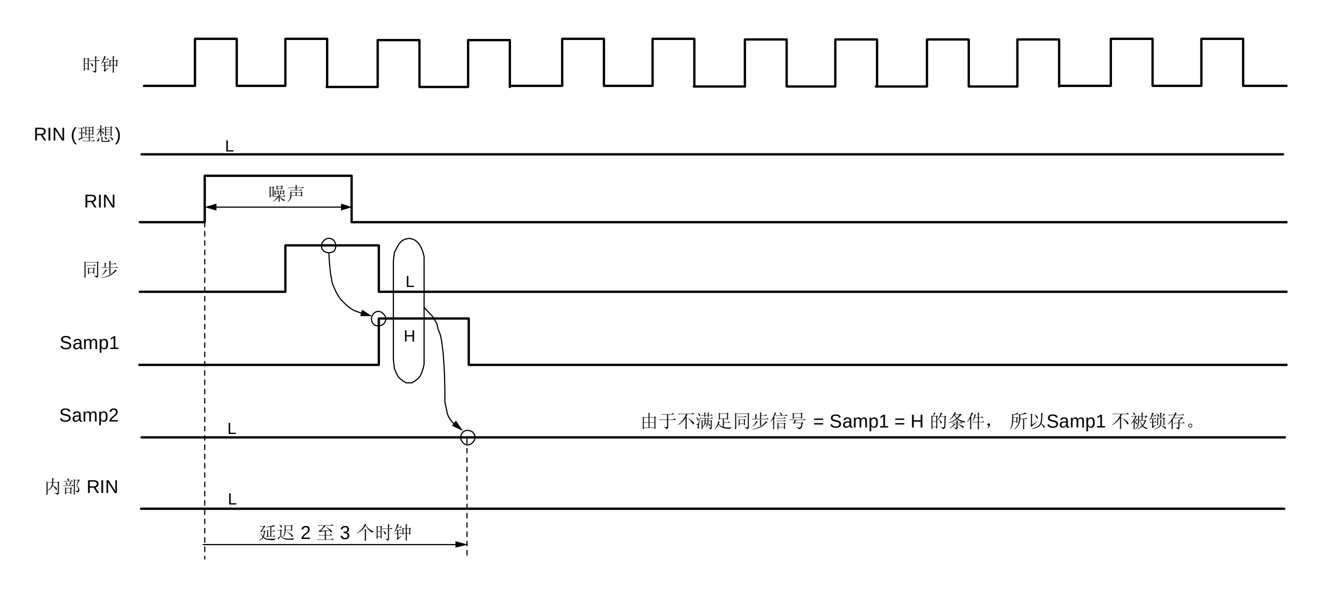
<5>　产生信号（samp3）对已经同步的信号和 samp2 进行第三次采样。
　（当已经同步的信号= samp2 = H，samp2 被锁存。）

<6>　使用 NCW 选择电路中作为 RIN 输入的信号。
　NCW = 0：　samp2 被处理作为电路中 RIN 引脚的输入。
　NCW = 1：　samp3 被处理作为电路中 RIN 引脚的输入。

图 19-16 显示噪声消除操作的示例。

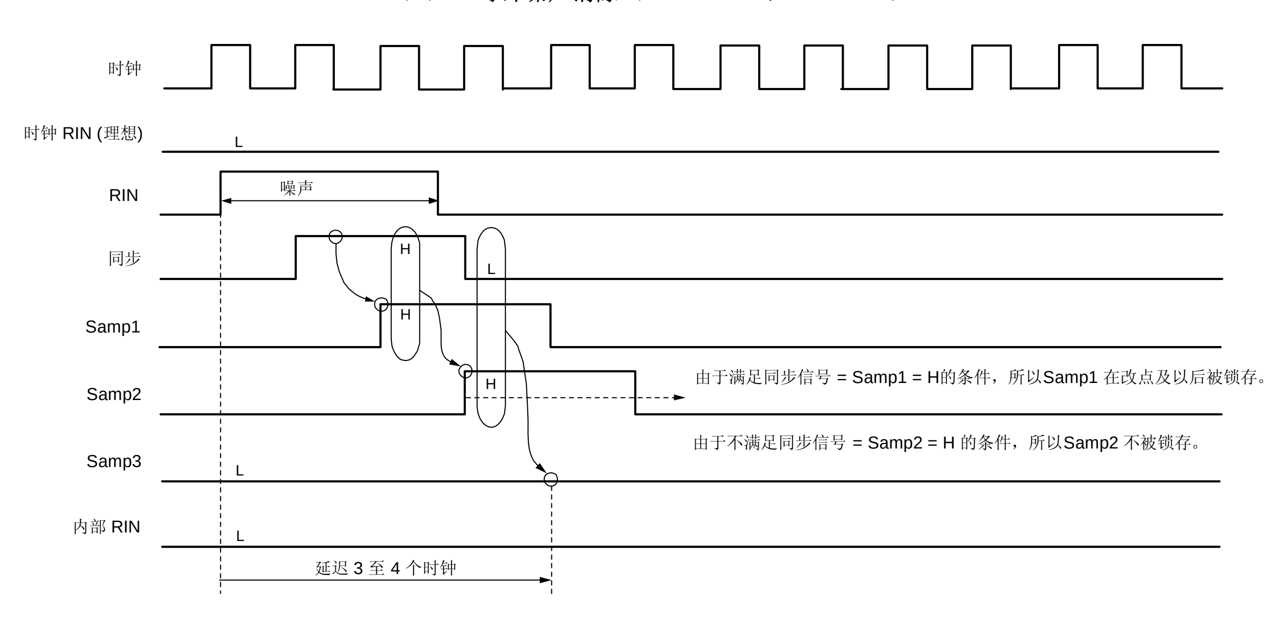
图 19-16. 噪声消除操作示例（1/2）

（a） 1 时钟噪声消除（PRSEN = 0，NCW = 0）



备注　内部 RIN 是一个同步后并执行两次采样的信号，所以要比实际从 RIN 引脚输入的外部信号晚 2 到 3 个时钟。

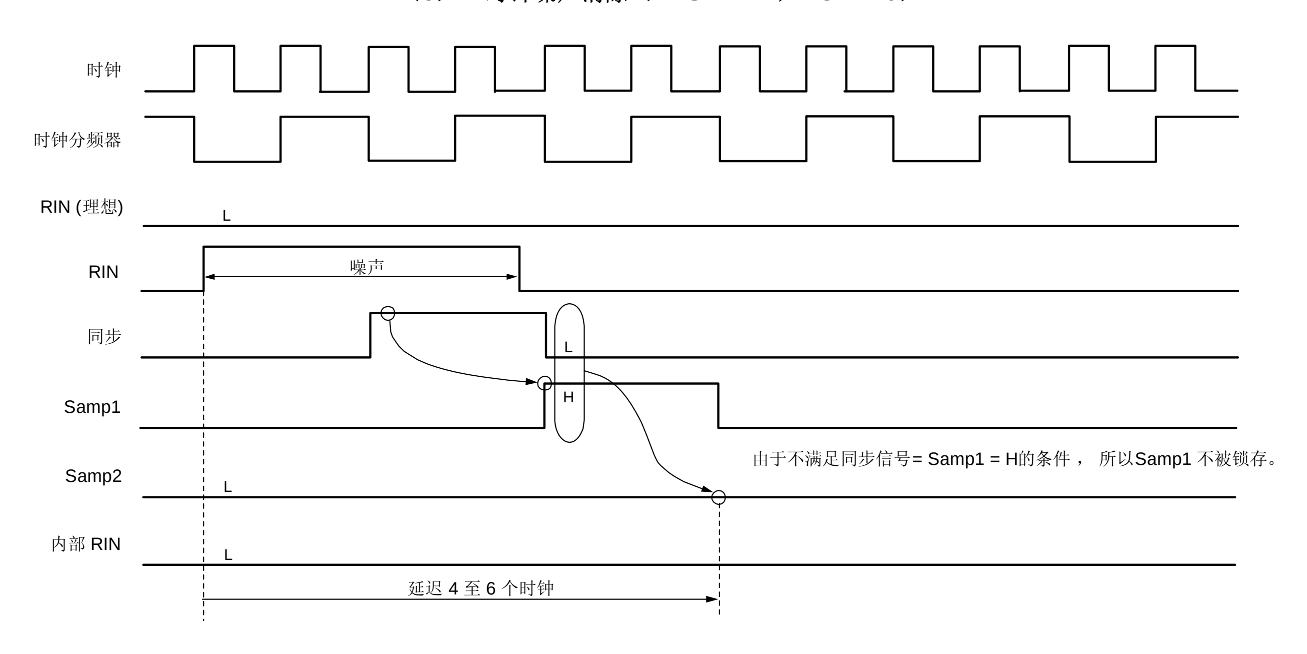
（b） 2 时钟噪声消除（PRSEN = 0，NCW = 1）



备注　内部 RIN 是一个同步后并执行三次采样的信号，所以要比实际从 RIN 引脚输入的外部信号晚 3 到 4 个时钟。

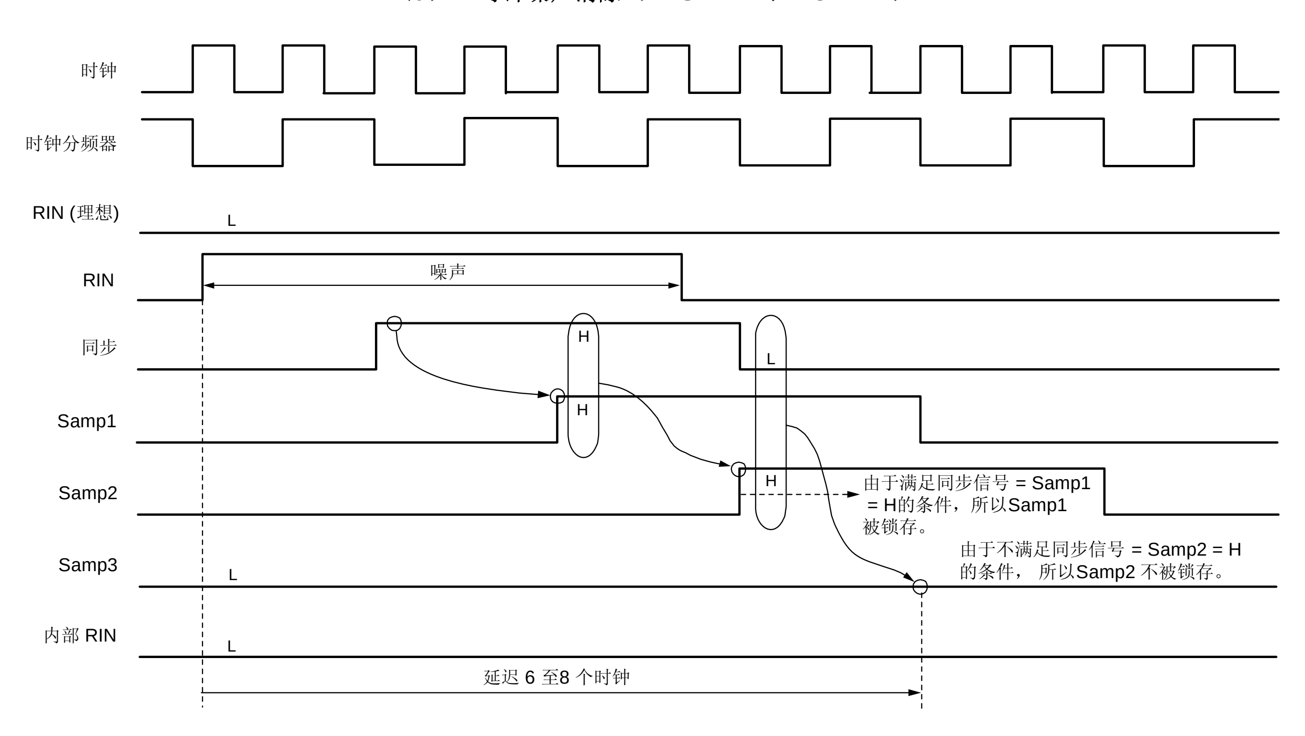
图 19-16. 噪声消除操作示例（2/2）

（c） 2 时钟噪声消除（PRSEN = 1，NCW = 0）



备注　内部 RIN 是一个同步后并执行两次采样的信号，所以要比实际从 RIN 引脚输入的外部信号晚 4 到 6 个时钟。

（d） 4 时钟噪声消除（PRSEN = 1，NCW = 1）



备注　内部 RIN 是一个同步后并执行两次采样的信号，所以要比实际从 RIN 引脚输入的外部信号晚 6 到 8 个时钟。

第二十章　中断功能

20.1 中断功能的类型

可以使用下列两种类型的中断功能。

（1）　可屏蔽中断

这类中断可进行屏蔽控制。通过设置优先级指定标志寄存器（PR0L，PR0H，PR1L，PR1H）将可屏蔽中断分为高中断优先级组和低中断优先级组。

支持多重中断，即在低优先级的中断服务程序中可以响应高优先级的中断请求。如果同时产生两个或两个以上具有相同优先级的中断请求，则根据向量中断服务的优先级进行处理。优先级的次序，参见**表 20-1**。

产生一个待机释放信号，并释放 STOP 和 HALT 模式。

可屏蔽中断包括外部中断请求和内部中断请求。

- μPD78F044x
  外部：7，内部：19
- μPD78F045x
  外部：7，内部：20
- μPD78F046x
  外部：7，内部：21

（2）　软件中断

这是通过执行 BRK 指令产生的向量中断。即使中断被禁止时也可以响应这该中断。软件中断不接受中断优先级控制。

20.2 中断源及配置

μPD78F044x 共有 26 个中断源，μPD78F045x 共有 27 个中断源，μPD78F046x 共有 28 个中断源，包括可屏蔽的中断和软件中断。另外，他们都具有 4 种复位源（参见**表 20-1**）。

表 20-1. 中断源列表（1/2）

中断类型	默认优先级[注 1]	中断源		内部/外部	向量表地址	基本配置类型[注 2]
		名称	触发器			
可屏蔽的	0	INTLVI	低电压检测[注 3]	内部	0004H	（A）
	1	INTP0	引脚输入边沿检测	外部	0006H	（B）
	2	INTP1			0008H	
	3	INTP2			000AH	
	4	INTP3			000CH	
	5	INTP4			000EH	
	6	INTSRE6	产生 UART6 接收错误	内部	0012H	（A）
	7	INTSR6	UART6 接收的结束		0014H	
	8	INTST6	UART6 发送的 结束		0016H	
	9	INTCSI10/ INTST0	CSI10 通信结束的结束/ UART0 发送的结束		0018H	
	10	INTTMH1	TMH1 与 CMP01 匹配（已经指定比较寄存器）		001AH	
	11	INTTMH0	TMH0 与 CMP00 匹配（已经指定比较寄存器）		001CH	
	12	INTTM50	TM50 与 CR50 匹配（已经指定比较寄存器）		001EH	
	13	INTTM000	TM00 与 CR000 匹配（已经指定比较寄存器），TI010 引脚有效边沿检测（已经指定捕获寄存器）		0021H	
	14	INTTM010	TM00 与 CR010 匹配（已经指定比较寄存器），TI000 引脚有效边沿检测（已经指定捕获寄存器）		0022H	
	15	INTAD[注 5]	10 位逐次逼近型 A/D 转换结束		0024H	
	16	INTSR0	UART0 接收的结束或产生接收错误		0026H	
	17	INTRTC	实时计数器/闹钟匹配检测的固定周期信号		0028H	
	18	INTTM51[注 4]	TM51 与 CR51 匹配（已经指定比较寄存器）		002AH	
	19	INTKR	按键中断检测	外部	002CH	（C）
	20	INTRTCI	实时计数器的间隔信号检测	内部	002EH	（A）

注　**1.** 当两个或更多可屏蔽中断同时产生时，默认的优先级决定向量中断的处理顺序。零表示最高优先级，26 表示最低优先级。

2. 基本配置类型（A）至（D）对应于图 20-1 中的（A）至（D）。

3. 当低电压检测寄存器（LVIM）的第 1 位（LVIMD）被清除为 0 时。

4. 当 8 位定时器/事件计数器 51 和 8 位计数器 H1 用于载波发生器模式时，在产生 INTTM5H1 信号时的时序产生中断（请看 **图 8-15 传输时序**）。

5. 仅限μPD78F045x 和 78F046x。

表 20-1. 中断源列表（2/2）

中断类型	默认优先级[注1]	中断源		内部/外部	向量表地址	基本配置类型[注2]
		名称	触发器			
可屏蔽	21	INTDSAD[注4]	16 位 ΔΣ 型 A/D 转换的结束	内部	0030H	（A）
	22	INTTM52	TM52 与 CR52 匹配（已经指定比较寄存器）		0032H	
	23	INTTMH2	TMH2 与 CRH2 匹配（已经指定比较寄存器）		0034H	
	24	INTMCG	曼彻斯特码接收的结束		0036H	
	25	INTRIN	遥控器接收边沿检测		0038H	
	26	INTRERR/ INTGP/ INTREND/ INTDFULL	遥控器接收产生错误 检测到遥控器引导脉冲 遥控器数据接收完成 遥控器 8 位移位数据的读取请求		003AH	
软件	–	BRK	BRK 指令执行	–	003EH	（D）
复位	–	复位后	复位输入	–	0000H	–
		POC	上电清零			
		LVI	低电压检测[注3]			
		WDT	WDT 溢出			

注 **1.** 当两个或更多可屏蔽中断同时产生时，默认的优先级决定向量中断的处理顺序。零表示最高优先级，26 表示最低优先级。

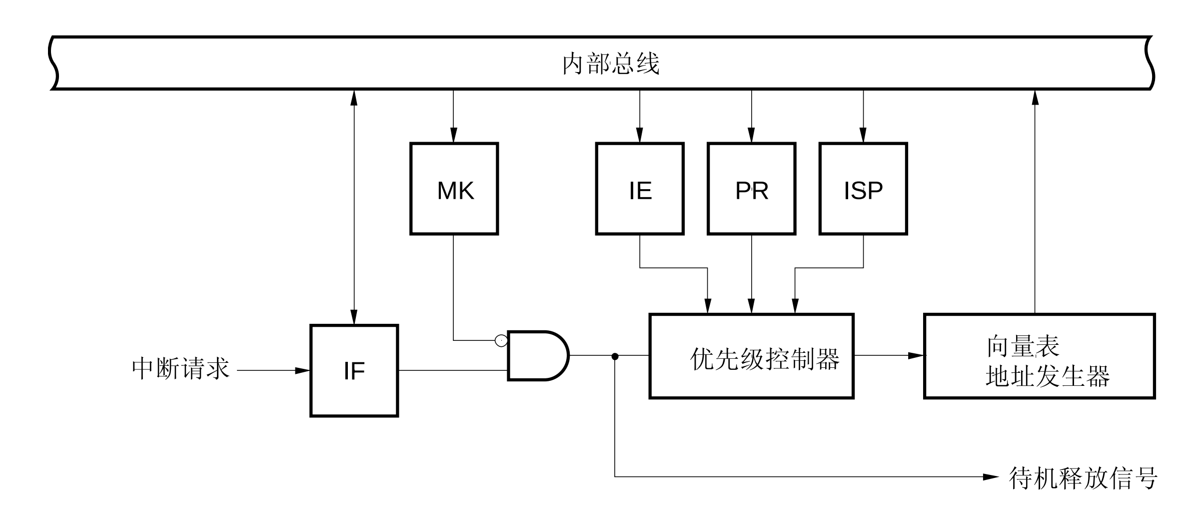
2. 基本配置类型（A）至（D）对应于图 20-1 中的（A）至（D）。

3. 当低电压检测寄存器（LVIM）的第 1 位（LVIMD）被设置为 1 时。

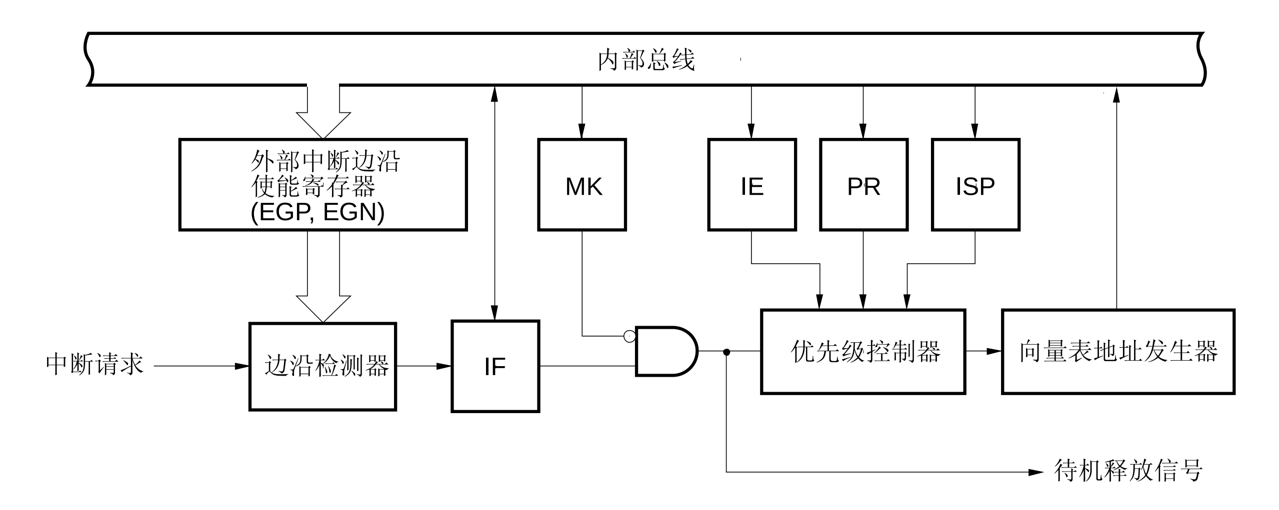
4. 仅限μPD78F046x。

图 **20-1.** 中断功能的基本配置（**1/2**）

（**A**）　内部可屏蔽中断



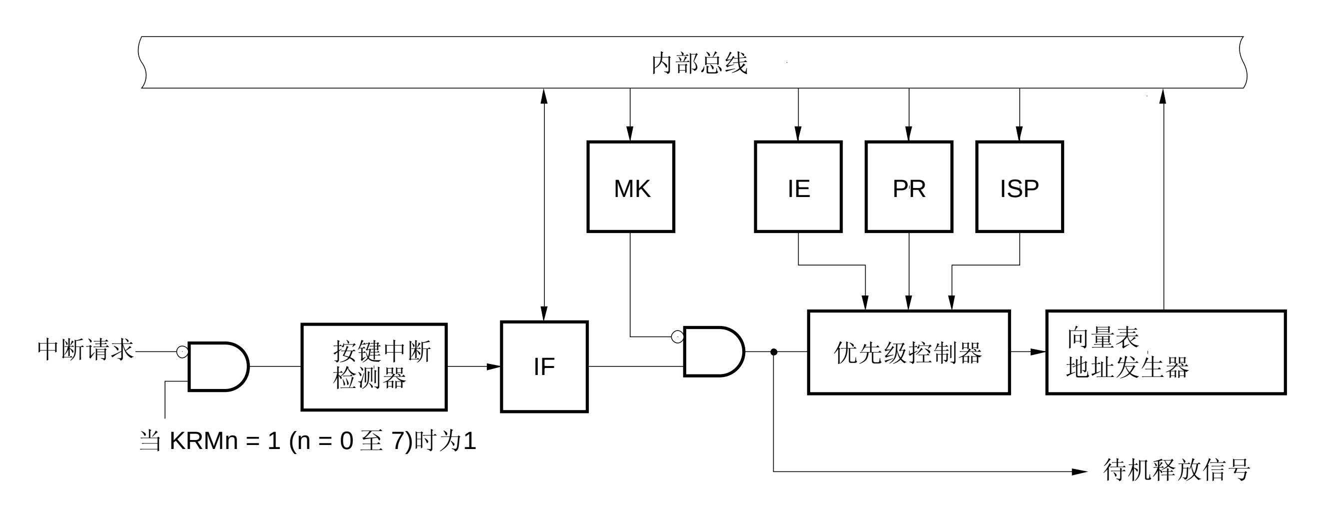
（**B**）　外部可屏蔽中断（**INTP0** 至 **INTP4**）



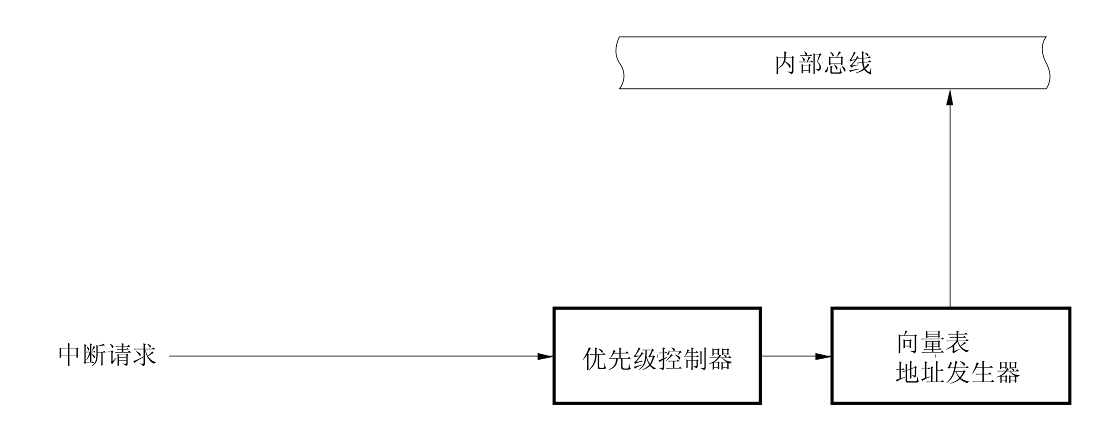
IF：　中断请求标志
IE：　中断使能标志
ISP：　正在服务的优先级标志
MK：　中断屏蔽标志
PR：　优先级指定标志

图 20-1. 中断功能的基本配置（2/2）

（C）　外部可屏蔽中断（INTKR）



（D）　软件中断



IF：　中断请求标志
IE：　中断允许标志
ISP：　服务优先级标志
MK：　中断屏蔽标志
PR：　优先级指定标志
KRM：　按键返回模式寄存器

20.3 控制中断功能的寄存器

以下 6 种类型的寄存器用于控制中断功能。

- 中断请求标志寄存器（IF0L、IF0H、IF1L、IF1H）
- 中断屏蔽标志寄存器（MK0L、MK0H、MK1L、MK1H）
- 优先级指定标志寄存器（PR0L、PR0H、PR1L、PR1H）
- 外部中断上升沿使能寄存器（EGP）
- 外部中断下降沿使能寄存器（EGN）
- 程序状态字（PSW）

表 20-2 列出了与中断请求源相对应的中断请求标志、中断屏蔽标志和优先级指定标志。

表 20-2. 与中断请求源相对应的标志（1/2）

中断源	中断请求标志		中断屏蔽标志		优先级指定标志	
		寄存器				寄存器
INTLVI	LVIIF	IF0L	LVIMK	MK0L	LVIPR	PR0L
INTP0	PIF0		PMK0		PPR0	
INTP1	PIF1		PMK1		PPR1	
INTP2	PIF2		PMK2		PPR2	
INTP3	PIF3		PMK3		PPR3	
INTP4	PIF4		PMK4		PPR4	
INTSRE6	SREIF6		SREMK6		SREPR6	
INTSR6	SRIF6	IF0H	SRMK6	MK0H	SRPR6	PR0H
INTST6	STIF6		STMK6		STPR6	
INTCSI10 INTST0	CSIIF10 注 1 STIF0 注 1		CSIMK10 注 2 STMK0 注 2		CSIPR10 注 3 STPR0 注 3	
INTTMH1	TMIFH1		TMMKH1		TMPRH1	
INTTMH0	TMIFH0		TMMKH0		TMPRH0	
INTTM50	TMIF50		TMMK50		TMPR50	
INTTM000	TMIF000		TMMK000		TMPR000	
INTTM010	TMIF010		TMMK010		TMPR010	

注　**1.** 如果产生中断源 INTCSI10 或 INTST0 中的任意一种，IF0H 的第 2 位被置位（1）。
　　2. MK0H 的第 2 位支持 INTCSI10 和 INTST0 两种中断源。
　　3. PR0H 的第 2 位支持 INTCSI10 和 INTST0 两种中断源。

表 **20-2.** 与中断请求源相对应的标志（**2/2**）

中断源	中断请求标志		中断屏蔽标志		优先级指定标志	
		寄存器				寄存器
INTAD 注 1	ADIF 注 1	IF1L	ADMK 注 1	MK1L	ADPR 注 1	PR1L
INTSR0	SRIF0		SRMK0		SRPR0	
INTRTC	RTCIF		RTCMK		RTCPR	
INTTM51 注 2	TMIF51		TMMK51		TMPR51	
INTKR	KRIF		KRMK		KRPR	
INTRTCI	RTCIIF		RTCIMK		RTCIPR	
INTDSAD 注 3	DSADIF 注 3		DSADMK 注 3		DASDPR 注 3	
INTTM52	TMIF52		TMMK52		TMPR52	
INTTMH2	TMHIF2	IF1H	TMHMK2	MK1H	TMHPR2	PR1H
INTMCG	MCGIF		MCGMK		MCGPR	
INTRIN	RINIF		RINMK		RINPR	
INTRERR INTGP INTREND INTDFULL	RERRIF 注 4 GPIF 注 4 RENDIF 注 4 DFULLIF 注 4		RERRMK 注 5 GPMK 注 5 RENDMK 注 5 DFULLMK 注 5		RERRPR 注 6 GPPR 注 6 RENDPR 注 6 DFULLPR 注 6	

注　**1.** 仅限μPD78F045x 和 78F046x。

2. 当 8 位定时器/事件计数器 51 和 8 位计数器 H1 用于载波发生器模式时，在产生 INTTM5H1 信号时的时序产生中断（请看 **图 8-15 传输时序**）。

3. 仅限μPD78F046x。

4. 如果产生中断源 INTRERR，INTGP，INTREND 或 INTDFULL 中的任意一种，IF1H 的第 3 位被置位（1）。

5. MK1H 的第 3 位支持 INTRERR，INTGP，INTREND 和 INTDFULL 中断源。

6. MK1H 的第 3 位支持 INTRERR，INTGP，INTREND 和 INTDFULL 中断源。

（1）　中断请求标志寄存器（IF0L，IF0H，IF1L，IF1H）

当产生对应的中断请求或执行一条指令时，中断请求标志被设置为 1。当执行的指令是响应中断请求或产生复位信号时，这些标志被清除为 0。

当响应中断时，中断请求标志被自动清除，然后进入中断服务程序。

可以通过 1 位或 8 位存储器操作指令来设置 IF0L、IF0H、IF1L 和 IF1H。当 IF0L 与 IF0H、IF1L 与 IF1H 组合为 16 位寄存器 IF0 和 IF1 时，可以通过 16 位存储器操作指令来设置这些寄存器。

复位信号的产生会将这些寄存器清除为 00H。

图 20-2. 中断请求标志寄存器（IF0L，IF0H，IF1L，IF1H）的格式

地址： FFE0H　复位后： 00H　R/W

符号	<7>	6	<5>	<4>	<3>	<2>	<1>	<0>
IF0L	SREIF6	0	PIF4	PIF3	PIF2	PIF1	PIF0	LVIIF

地址： FFE1H　复位后： 00H　R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF0H	TMIF010	TMIF000	TMIF50	TMIFH0	TMIFH1	CSIIF10 STIF0	STIF6	SRIF6

地址： FFE2H　复位后： 00H　R/W

<R>

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF1L	TMIF52	DSADIF[注 2]	RTCIIF	KRIF	TMIF51	RTCIF	SRIF0	ADIF[注 1]

地址： FFE3H　复位后： 00H　R/W

符号	7	6	5	4	<3>	<2>	<1>	<0>
IF1H	0	0	0	0	RERRIF GPIF RENIF DFULLIF	RINIF	MCGIF	TMHIF2

XXIFX	中断请求标志
0	没有产生中断请求信号
1	产生中断请求，中断请求状态

注　**1.** 仅限μPD78F045x 和 78F046x。
　　2. 仅限μPD78F046x。

注意事项　**1.** 请确保将 **IF0H** 的第 **6** 位和 **IF1H** 的第 **4** 位至第 **7** 位清除为 **0**。
　　　　　2. 当待机释放后要操作定时器、串行接口或 **A/D** 转换器时，先将中断请求标志清零后再操作。可能会因为噪音而置位中断请求标志。

注意事项 **3.** 当操作中断请求标志寄存器中的标志时，使用 **1** 位存储器操作指令（**CLR1**）。当使用 **C** 语言描述时**,** 因为编译后的汇编指令必须是 **1** 位存储器操作指令（**CLR1**），所以应该使用位操作指令，比如 **"IF0L.0 = 0;"** 或 **"_asm（"clr1 IF0L，0"）;"**。

如果 **C** 语言描述的程序，使用 **8** 位存储器操作指令比如**"IF0L &= 0xfe;"**，则编译后该语句将被转换为三条汇编指令：

```
mov a，IF0L
and a，#0FEH
mov IF0L，a
```

这种情况下，当处于**"mov a，IF0L"**和 **"mov IF0L，a"**之间的时序时，即使同一中断请求标志寄存器（**IF0L**）的另一位请求标志被设置为 **1**，则该请求标志也将在**"mov IF0L，a"**时被清除为 **0**。因此，在 **C** 语言中使用 **8** 位存储器操作指令时必须小心。

（2）　中断屏蔽标志寄存器（MK0L，MK0H，MK1L，MK1H）

这些中断屏蔽标志用于使能/禁止对应的可屏蔽中断服务。

可以通过 1 位或 8 位存储器操作指令来设置 MK0L、 MK0H、MK1L 和 MK1H。 当 MK0L 与 MK0H，MK1L 与 MK1H 组合为 16 位寄存器 MK0 与 MK1 时，可以通过 16 位存储器操作指令来设置这些寄存器。

复位信号的产生会将这些寄存器设置为 FFH。

图 20-3. 中断屏蔽标志寄存器（MK0L，MK0H，MK1L，MK1H）的格式

地址：FFE4H　复位后：FFH　R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
MK0L	SREMK6	1	PMK4	PMK3	PMK2	PMK1	PMK0	LVIMK

地址：FFE5H　复位后：FFH　R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
MK0H	TMMK010	TMMK000	TMMK50	TMMKH0	TMMKH1	CSIMK10 STMK0	STMK6	SRMK6

地址：FFE6H　复位后：FFH　R/W

<R>

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
MK1L	TMMK52	DASDMK [注2]	RTCIMK	KRMK	TMMK51	RTCMK	SRMK0	ADMK [注1]

地址：FFE7H　复位后：FFH　R/W

符号	7	6	5	4	<3>	<2>	<1>	<0>
MK1H	1	1	1	1	RERRMK GPMK RENDMK DFULLMK	RINMK	MCGMK	TMHMK2

XXMKX	中断服务控制
0	使能中断服务
1	禁止中断服务

注　**1.** 仅限μPD78F045x 和 78F046x。

　　2. 仅限μPD78F046x。

注意事项　请确保将 MK0H 的第 6 位和 MK1H 的第 4 位至第 7 位设置为 1。

（3）　优先级指定标志寄存器（PR0L，PR0H，PR1L，PR1H）

这些优先级指定标志寄存器用于设置相关的可屏蔽中断优先级次序。

可由 1 位或 8 位存储器操作指令设置 PR0L、PR0H、PR1L 和 PR1H。当 PR0L 与 PR0H，PR1L 与 PR1H 组合起来形成 16 位寄存器 PR0 与 PR1 时，可由 16 位存储器操作指令设置这些寄存器。

复位信号的产生将这些寄存器内容设置为 FFH。

图 20-4. 优先级指定标志寄存器（PR0L，PR0H，PR1L，PR1H）的格式

地址：FFE8H　复位后：FFH　R/W

符号	<7>	6	<5>	<4>	<3>	<2>	<1>	<0>
PR0L	SREPR6	1	PPR4	PPR3	PPR2	PPR1	PPR0	LVIPR

地址：FFE9H　复位后：FFH　R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR0H	TMPR010	TMPR000	TMPR50	TMPRH0	TMPRH1	CSIPR10 STPR0	STPR6	SRPR6

符号：FFEAH　复位后：　FFH　R/W

<R>

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR1L	TMPR52	DSADPR 注 2	RTCIPR	KRPR	TMPR51	RTCPR	SRPR0	ADPR 注 1

地址：FFEBH　复位后：FFH　R/W

符号	7	6	5	4	<3>	<2>	<1>	<0>
PR1H	1	1	1	1	RERRPR GPPR RENDPR DFULLPR	RINPR	MCGPR	TMHPR2

XXPRX	优先级选择
0	高优先级
1	低优先级

注　**1.** 仅限μPD78F045x 和 78F046x。
　　2. 仅限μPD78F046x。

注意事项　请确保将 PR0H 的第 6 位和 PR1H 的第 4 位至第 7 位设置为 1。

（4）　外部中断上升沿使能寄存器（EGP），外部中断下降沿使能寄存器（EGN），

这两个寄存器用于指定 INTP0 至 INTP4 的有效边沿。

可以通过 1 位或 8 位存储器操作指令来设置 EGP 和 EGN。

复位信号的产生将这些寄存器内容清除为 00H。

图 20-5. 外部中断上升沿使能寄存器（EGP）和外部中断下降沿使能寄存器（EGN）的格式

地址：FF48H　复位后：00H　R/W

符号	7	6	5	4	3	2	1	0
EGP	0	0	0	EGP4	EGP3	EGP2	EGP1	EGP0

地址：FF49H　复位后：00H　R/W

符号	7	6	5	4	3	2	1	0
EGN	0	0	0	EGN4	EGN3	EGN2	EGN1	EGN0

EGPn	EGNn	INTPn 引脚有效边沿选择（n = 0 至 4）
0	0	禁止边沿检测
0	1	下降沿
1	0	上升沿
1	1	兼有上升和下降双沿

表 20-3 展示了与 EGPn 和 EGNn 对应的端口。

表 20-3. 与 EGPn 和 EGNn 对应的端口

检测允许寄存器		边沿检测端口	中断请求信号
EGP0	EGN0	P120/EXLVI	INTP0
EGP1	EGN1	P34/TI52/TI010/TO00/RTC1HZ	INTP1
EGP2	EGN2	P33/TI000/RTCDIV/RTCCL/BUZ	INTP2
EGP3	EGN3	P31/TOH1	INTP3
EGP4	EGN4	P14	INTP4

注意事项　**通过将 EGPn 和 EGNn 清除为 0 来选择端口模式。因为外部中断功能切换到端口功能时，可能会检测到边沿。**

备注　n = 0 至 4

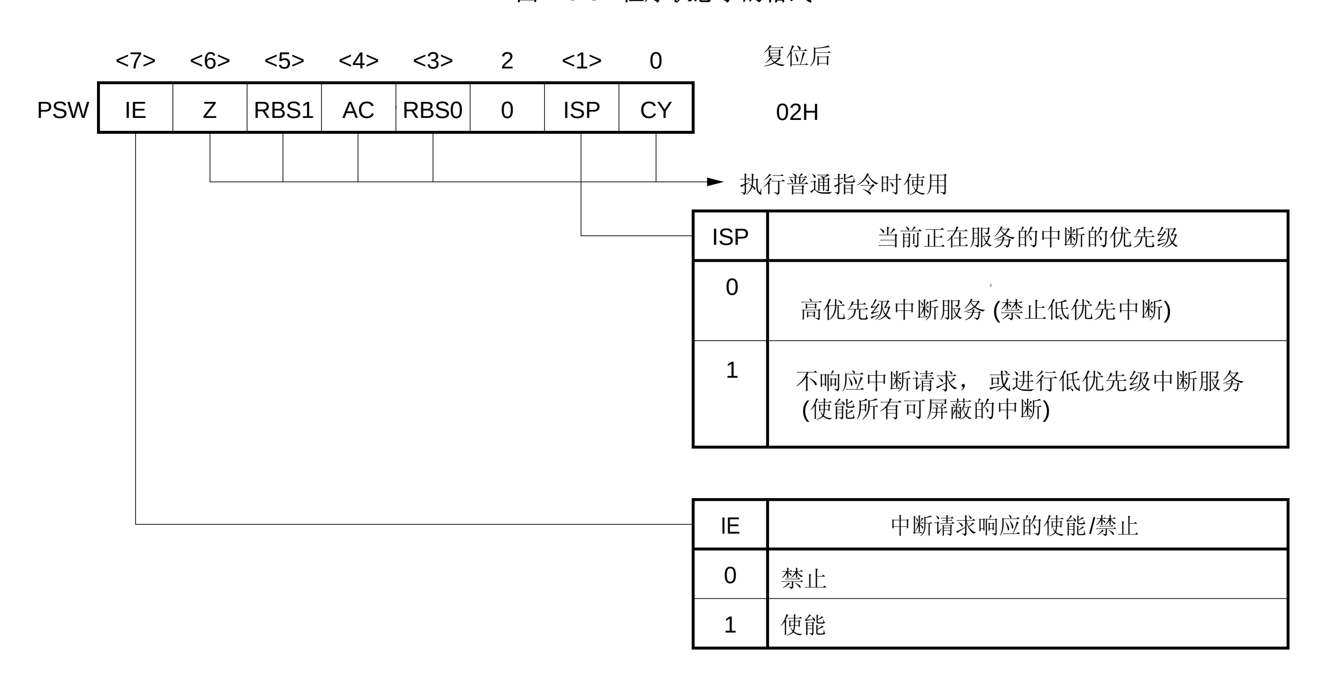
（5） 程序状态字（PSW）

程序状态字是用于保存指令执行结果和中断请求当前状态的寄存器。设置使能/禁止可屏蔽中断的 IE 标志和控制多重中断服务的 ISP 标志，都被映射到 PSW 中。

除了 8 位读/写之外，还可使用位操作指令和专用指令（EI 和 DI）对该寄存器进行操作。在响应向量中断请求时，如果执行 BRK 指令，则将 PSW 的内容自动保存到堆栈中，并且将 IE 标志重置为 0。如果响应可屏蔽中断请求，则将被响应中断的优先级指定标志的内容传输到 ISP 标志。使用 PUSH PSW 指令也会将 PSW 的内容保存到堆栈中。而使用 RETI、RETB 和 POP PSW 指令可以他们从堆栈恢复。

复位信号的产生会将 PSW 设置为 02H。

图 20-6. 程序状态字的格式



20.4 中断服务操作

20.4.1 可屏蔽的中断响应

当中断请求标志被设置为 1 且对应该中断请求的屏蔽标志（MK）被清除为 0 时，可以响应这个可屏蔽中断。如果中断处于中断使能状态（当 IE 表示被设置为 1），可以响应向量中断请求。但是，在一个较高优先级中断请求服务期间（当 ISP 标志被重置为 0），低优先级的中断请求不被响应。

从一个可屏蔽中断请求的产生到执行矢量中断服务的时间在表 20-4 中列出。

中断请求响应时序，参见图 **20-8** 和 **20-9**。

表 20-4. 从可屏蔽中断的产生到服务所需要的时间

	最短时间	最长时间[注]
当 ××PR = 0 时	7 个时钟	32 个时钟
当 ××PR = 1 时	8 个时钟	33 个时钟

注 如果是紧邻在除法指令执行之前产生一个中断请求，则等待时间会较长。

备注 1 个时钟：$1/f_{CPU}$（f_{CPU}： CPU 时钟）

如果同时产生两个或更多的可屏蔽中断请求，则首先响应优先级指定标志中被指定为较高优先级的请求。如果两个或更多的中断请求具有相同的优先级，则首先响应具有最高默认优先级的请求。

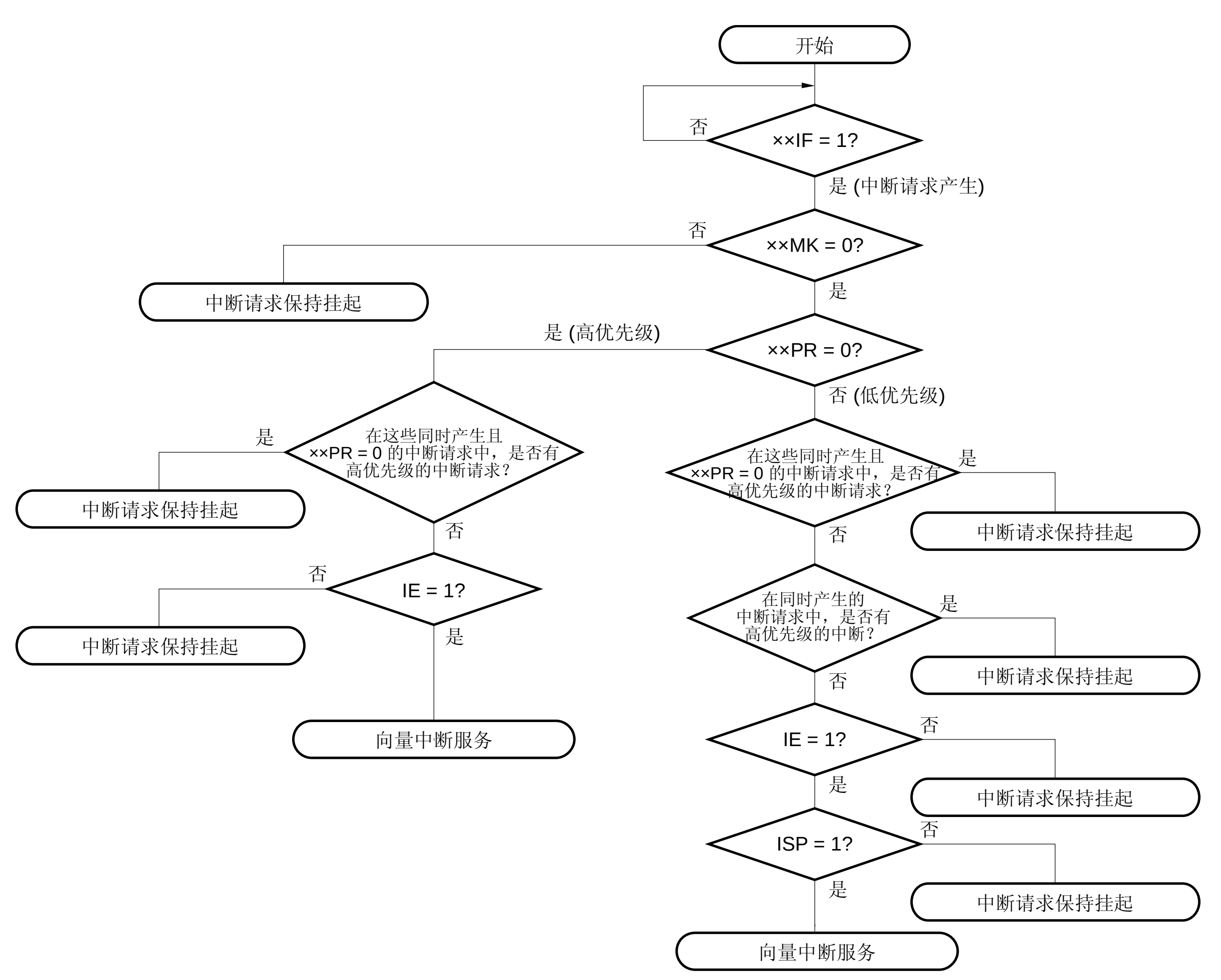
当被挂起的中断请求变为可响应时，它可以被响应。

图 20-7 展示了中断请求响应算法。

如果响应了一个可屏蔽中断请求，则按照先 PSW 再 PC 的顺序将他们的内容依次保存到堆栈中，IE 标志被重置为 0，并将被响应的中断对应的优先级指定标志的内容传送到 ISP 标志。由各个中断请求决定的向量表数据被装载到 PC 中，并跳转。

使用 RETI 指令可以从中断恢复。

图 20-7. 中断请求响应处理算法



××IF：　中断请求标志

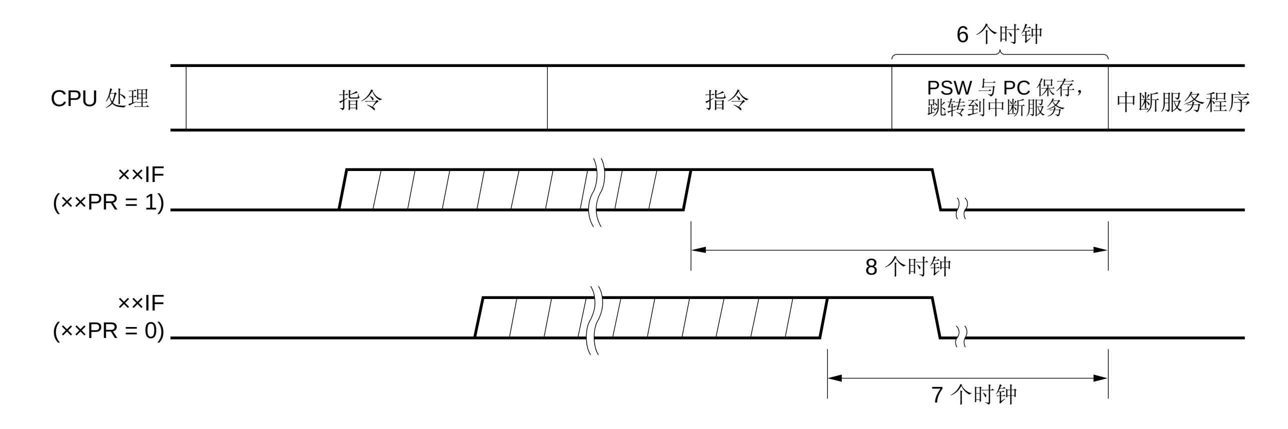
××MK：　中断屏蔽标志

××PR：　优先级指定标志

IE：　控制可屏蔽中断请求的响应的标志 （1 = 使能、 0 = 禁止）

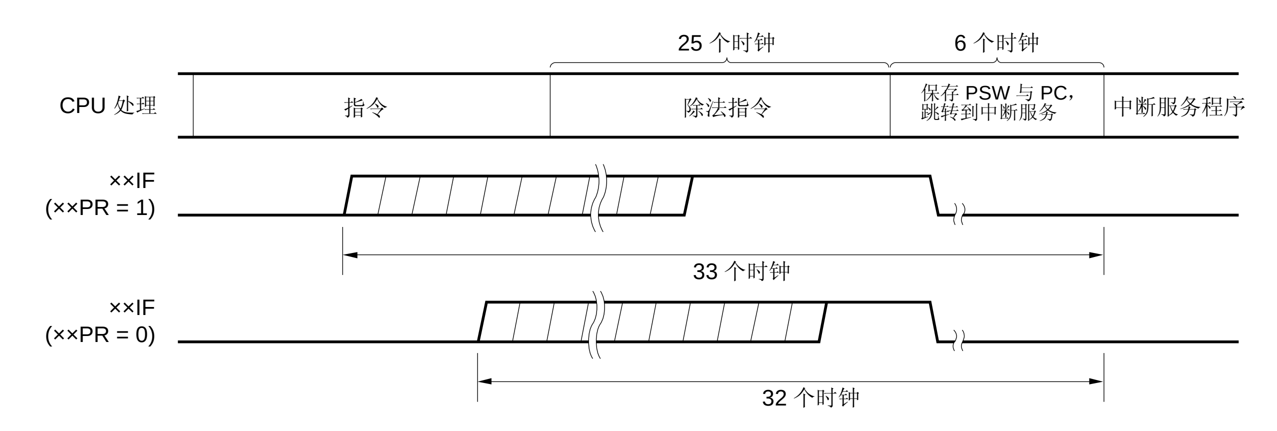
ISP：　指示正在服务的中断优先级的标志 （0 = 高优先级中断正在服务中，1 = 没有中断请求被响应，或低优先级正在服务中）

图 **20-8.** 中断请求响应时序（最短时间）



备注　1 个时钟：1/f_{CPU}（f_{CPU}：CPU 时钟）

图 **20-9.** 中断请求响应时序（最长时间）



备注　1 个时钟：1/f_{CPU}（f_{CPU}：CPU 时钟）

20.4.2 软件中断请求响应

执行 BRK 指令可响应软件中断。软件中断不能被禁止。

如果响应一个软件中断请求，则按照先程序状态字（PSW）再程序计数器（PC）的顺序将他们的内容依次保存到堆栈中，IE 标志被重置（0）。向量表（003EH，003FH）的内容被装载到 PC 中，并跳转。

使用 RETB 指令可以从软件中断恢复。

注意事项　不能使用 RETI 指令从软件中断恢复。

20.4.3 多重中断服务

在一个中断服务程序执行期间，又响应了另一个中断请求，这时发生多重中断服务。

除非选择中断请求响应使能状态（IE = 1），否则不会发生多重中断服务。在一个中断请求已经被响应时，中断请求响应变为禁止（IE = 0）。因此，如果要使能多重中断服务，必需在中断服务期间使用 EI 指令将 IE 标志置位（1），从而使能中断响应。

此外，即使中断被使能，多重中断服务可能没有被使能，这是因为受到中断优先级的控制。可以使用两类优先级控制：默认优先级控制和可编程优先级控制。可编程优先级控制用于多重中断服务。

在中断使能状态中，如果产生的中断请求的优先级等于或高于当前正在服务的中断的优先级，则响应该中断请求，发生多重中断服务。如果产生的中断请求的优先级低于当前正在服务的中断的优先级，则不响应该中断请求，不发生多重中断服务。由于中断处于中断禁止状态或优先级较低，未使能的中断请求被挂起。当前的中断服务结束时，则在执行至少一条主程序指令后，才可响应被挂起的中断请求。

表 20-5 展示了为多重中断服务使能的中断请求之间的关系，图 20-10 为多重中断服务示例。

表 20-5. 中断服务期间为多重中断服务使能的中断请求之间的关系

多重中断服务 / 正在服务的中断		可屏蔽中断请求				软件中断请求
		PR = 0		PR = 1		
		IE = 1	IE = 0	IE = 1	IE = 0	
可屏蔽的中断	ISP = 0	○	×	×	×	○
	ISP = 1	○	×	○	×	○
软件中断		○	×	○	×	○

备注　**1.** ○：　使能多重中断服务

2. ×：　禁止多重中断服务

3. ISP 和 IE 时包含在 PSW 中的标志位。

ISP = 0：　正在服务的中断具有较高优先级。

ISP = 1：　没有响应中断请求，或正在服务的中断具有较低优先级。

IE = 0：　禁止响应中断请求。

IE = 1：　使能响应中断请求。

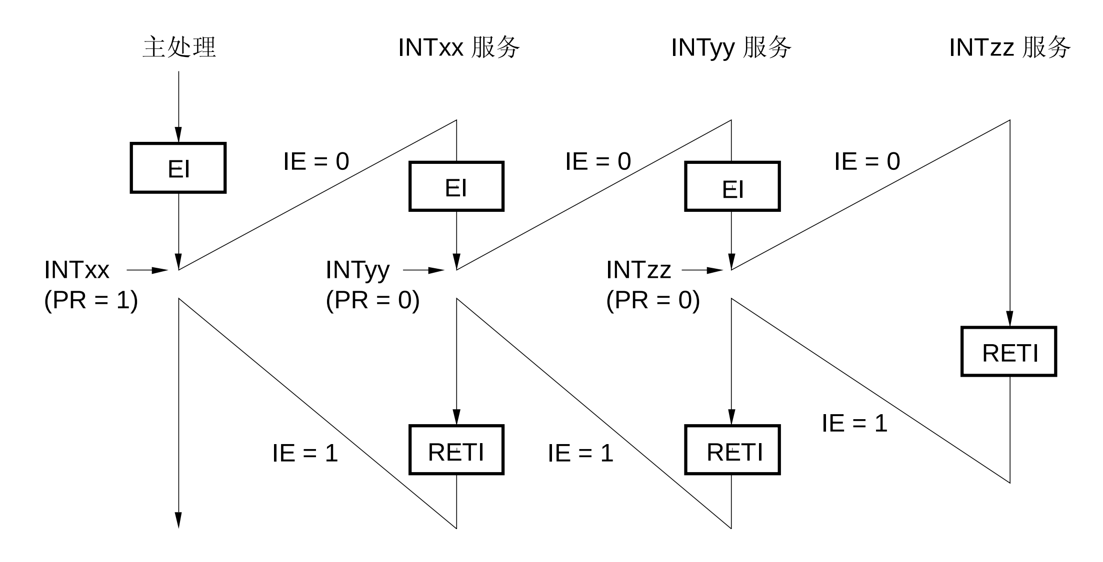
4. PR 是包含在 PR0L、PR0H、PR1L 和 PR1H 中的标志位。

PR = 0：　高优先级

PR = 1：　低优先级

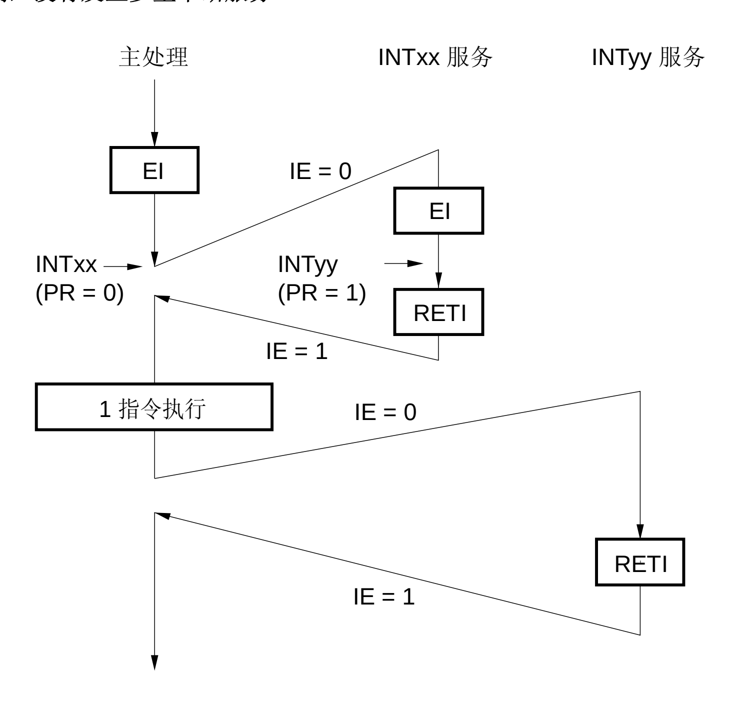
图 **20-10.** 多重中断服务示例（**1/2**）

例 1. 发生两次多重中断服务



在中断 INTxx 的服务期间，响应了两个中断请求 INTyy 和 INTzz，发生了多重中断服务。在响应每个中断之前，总是必须执行 EI 指令来使能中断请求的响应。

例 2. 因为优先级控制，没有发生多重中断服务

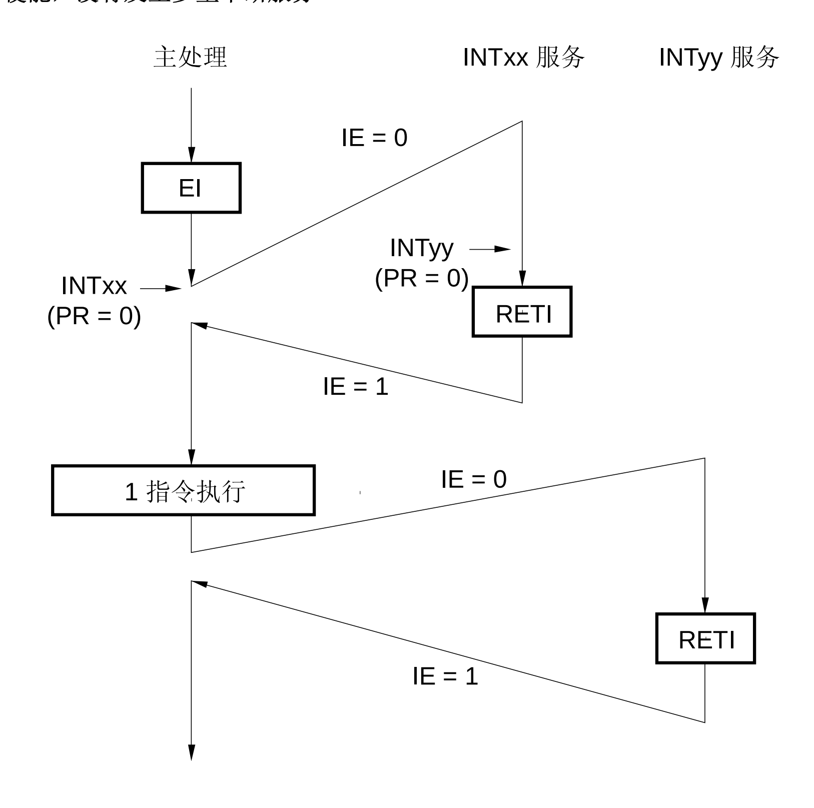


在 INTxx 中断服务期间产生的中断请求 INTyy 不被响应，因为 INTyy 的优先级低于 INTxx 的优先级，不会发生多重中断服务。中断请求 INTyy 被挂起，在执行一条主程序指令后，响应该中断请求。

PR = 0： 高优先级
PR = 1： 低优先级
IE = 0： 禁止响应中断请求

图 **20-10.** 多重中断服务示例（**2/2**）

例 **3.** 因为中断未使能，没有发生多重中断服务



在中断 INTxx 的服务期间，不使能中断（未执行 EI 指令），因此，不响应中断请求 INTyy，不会发生多重中断服务。中断请求 INTyy 被挂起，在执行一条主程序指令后，响应该中断请求。

PR = 0：　高优先级
IE = 0：　禁止响应中断请求

20.4.4 中断请求保持

在某些指令的执行期间，即使出现中断请求，请求响应被挂起，直到下一条指令执行结束。以下列出这这些指令（中断请求保持指令）。

- MOV PSW, #byte
- MOV A, PSW
- MOV PSW, A
- MOV1 PSW. bit, CY
- MOV1 CY, PSW. bit
- AND1 CY, PSW. bit
- OR1 CY, PSW. bit
- XOR1 CY, PSW. bit
- SET1 PSW. bit
- CLR1 PSW. bit
- RETB
- RETI
- PUSH PSW
- POP PSW
- BT PSW. bit, $addr16
- BF PSW. bit, $addr16
- BTCLR PSW. bit, $addr16
- EI
- DI
- 用于 IF0L，IF0H，IF1L，IF1H，MK0L，MK0H，MK1L，MK1H，PR0L，PR0H，PR1L 和 PR1H 寄存器的操作指令。

注意事项　BRK 指令不属于上述列出的中断请求保持指令。但是，通过执行 BRK 指令激活的软件中断会清除 IE 标志。因此，即使在执行 BRK 指令期间产生可屏蔽中断请求，该中断请求也不会被响应。

图 20-11 展示了中断请求被挂起的时序。

图 20-11. 中断请求保持

CPU 处理	指令 N	指令 M	保存PSW 与 PC，跳转到中断服务	中断服务程序

××IF

备注
1. 指令 N：　中断请求保持指令
2. 指令 M：　除中断请求保持指令之外的指令
3. ××PR（优先级）的值不会影响××IF （中断请求）的操作。

第二十一章　按键中断功能

21.1 按键中断的功能

设置按键返回模式寄存器（KRM）并有一个下降沿输入到按键中断输入引脚（KR0 至 KR4），可以产生按键中断（INTKR）。

表 21-1. 按键中断检测引脚的分配

标志	描述
KRM0	以 1 位单元控制 KR0 信号。
KRM1	以 1 位单元控制 KR1 信号。
KRM2	以 1 位单元控制 KR2 信号。
KRM3	以 1 位单元控制 KR3 信号。
KRM4	以 1 位单元控制 KR4 信号。

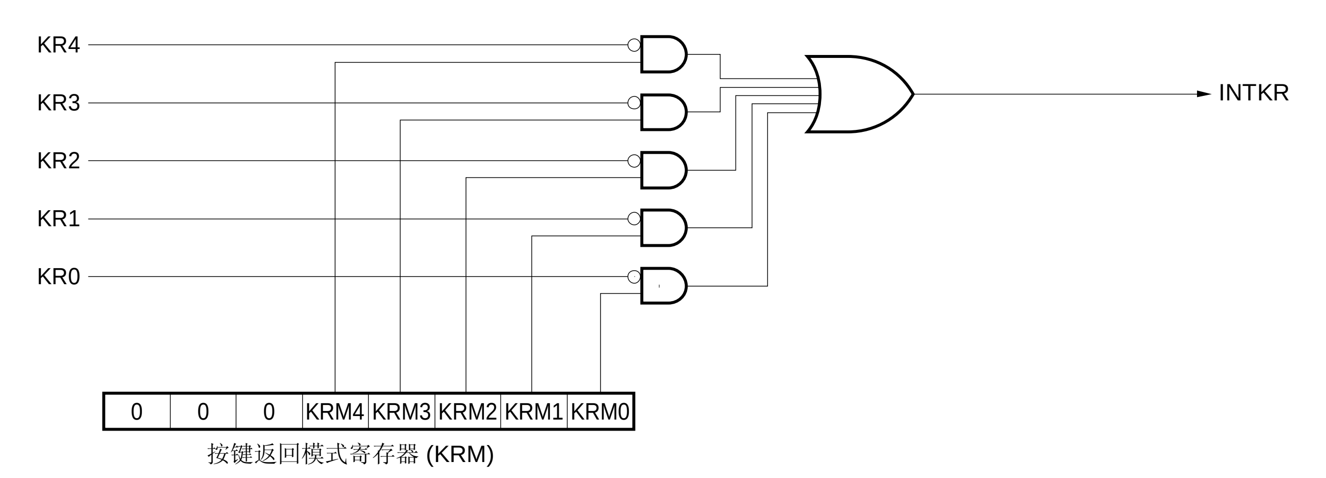
21.2 按键中断的配置

按键中断包括以下硬件。

表 21-2. 按键中断的配置

项目	配置
控制寄存器	按键返回模式寄存器（KRM）

图 21-1. 按键中断的框图



21.3 控制按键中断的寄存器

（1） 按键返回模式寄存器（KRM）

该寄存器用于设置当产生一个按键中断输入引脚（KR0 至 KR4）下降沿时是否检测按键中断（INTKR）。

可以通过 1 位或 8 位存储器操作指令来设置 KRM。

复位信号的产生会将 KRM 清除为 00H。

图 21-2. 按键返回模式寄存器 （KRM） 的格式

地址： FF6EH　复位后： 00H　R/W

符号	7	6	5	4	3	2	1	0
KRM	0	0	0	KRM4	KRM3	KRM2	KRM1	KRM0

KRMn	按键中断模式控制 (n = 0至4)
0	不检测按键中断信号
1	检测按键中断信号

注意事项

1. **设置 KRMn 位为 1 来使用按键中断功能时，设置上拉电阻寄存器 4（PU4）的对应的 PU4n 位为 1。**
2. **如果 KRM 被改变，则中断请求标志可能被置位。因此，先禁止中断，然后修改 KRM 寄存器。清除中断请求标志并使能中断。**
3. **如果按键中断信号的检测被禁止（KRMn = 0），对应的 P4n 引脚可以用作普通端口。**
4. **要使用 P40/KR0/V_{LC3} 引脚作为按键中断功能（KR0），将 LCD 显示模式寄存器（LCDM）设置为 1/4 偏压方式之外。如果 P40/KR0/V_{LC3} 引脚被设置为 1/4 偏压方式，它被作为 V_{LC3}。**
5. <R> **当使用 segment 键扫描输入(KSON = 1)功能时，KRn 引脚被用作 segment 键扫描输入引脚，设置 KRMn 为 1。KRn 引脚不被用作 segment 键扫描输入引脚，设置 KRMn 为 0。（参见图 17-7）**

第二十二章　待机模式

22.1 待机功能及配置

22.1.1 待机功能

待机功能的设计是用来降低系统的工作电流，有以下两种模式可供使用。

（1）　**HALT 模式**

通过执行 HALT 指令设置 HALT 模式。在 HALT 模式下，CPU 操作时钟停止。如果设置 HALT 模式前，高速系统时钟振荡器、内部高速振荡器、内部低速振荡器或副系统时钟振荡器正在工作，则各种时钟继续振荡。在该模式下，工作电流的减少没有 STOP 模式那么多，但 HALT 模式对于中断请求产生后立即重启的操作以及频繁执行间歇操作非常有效。

（2）　**STOP 模式**

通过执行 STOP 指令设置 STOP 模式。在 STOP 模式中，高速系统时钟振荡器和内部高速振荡器停止操作，整个系统停止，因此，CPU 的工作电流将会大幅下降。

可以通过中断请求释放该模式，这样可以使能执行间歇操作。但是在选择 X1 时钟时，释放 STOP 模式之后需要等待一段时间，来确保振荡器振荡稳定时间，因此如果需要在产生中断请求后立即进行处理，则应选择 HALT 模式。

在这两种模式下，寄存器、标志和数据存储器的所有内容将会保持进入待机模式前的瞬时值。I/O 端口输出锁存器和输出缓冲器的状态也将被保持。

注意事项 **1. 仅当 CPU 运行于主系统时钟时，才能使用 STOP 模式。副系统时钟的振荡不能停止。当 CPU 运行于主系统时钟或副系统时钟时，可以使用 HALT 模式。**

2. 当切换到 STOP 模式时，在执行 STOP 指令之前，请确保停止那些使用主系统时钟的外设硬件的操作。

3. 当使用待机功能时，推荐采用以下步骤来降低 10 位逐次逼近型 A/D 转换器的操作电流：首先将 A/D 转换器模式寄存器（ADM）的第 7 位（ADCS）和第 0 位（ADCE）清除为 0，停止 A/D 转换操作，然后执行 STOP 指令。

当使用待机模式时，推荐采用以下步骤来降低 16 位ΔΣ型 A/D 转换器的操作电流：首先将 16 位ΔΣ型 A/D 转换器模式寄存器的第 7 位（ADDPON）和第 6 位（ADDCE）清除为 0，停止 A/D 转换器操作，然后执行 STOP 指令。

22.1.2 控制待机功能的寄存器

待机功能由以下两个寄存器控制。

- 振荡稳定时间计数器的状态寄存器（OSTC）
- 振荡稳定时间选择寄存器（OSTS）

备注 启动、停止以及时钟选择的寄存器，参见 **第五章 时钟发生器。**

（1） 振荡稳定时间计数器的状态寄存器（OSTC）

该寄存器用于指示 X1 时钟振荡稳定时间计数器的计数状态。如果通过用作 CPU 时钟的内部高速振荡时钟或副系统时钟来启动 X1 振荡时钟，则可以检查 X1 时钟振荡稳定时间。

可以通过 1 位或 8 位存储器操作指令来读取 OSTC。

复位释放（通过 $\overline{\text{RESET}}$ 输入、POC、LVI 和 WDT 复位）时、STOP 指令以及 MSTOP（MOC 寄存器的第 7 位）= 1 可以将 OSTC 清除为 00H。

图 **22-1. 振荡稳定时间计数器状态寄存器 （OSTC）的格式**

地址： FFA3H 复位后： 00H R

符号	7	6	5	4	3	2	1	0
OSTC	0	0	0	MOST11	MOST13	MOST14	MOST15	MOST16

MOST11	MOST13	MOST14	MOST15	MOST16	振荡稳定时间的状态	
						f_X = 10 MHz
1	0	0	0	0	$2^{11}/f_X$ 最小值	204.8 μs 最小值
1	1	0	0	0	$2^{13}/f_X$ 最小值	819.2 μs 最小值
1	1	1	0	0	$2^{14}/f_X$ 最小值	1.64 ms 最小值
1	1	1	1	0	$2^{15}/f_X$ 最小值	3.27 ms 最小值
1	1	1	1	1	$2^{16}/f_X$ 最小值	6.55 ms 最小值

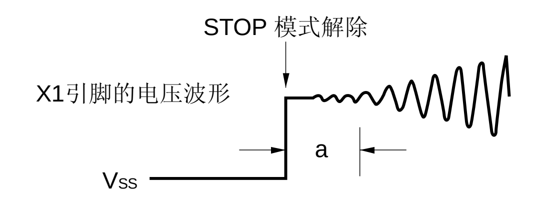
注意事项 **1.** 在经过上述时间后，从 **MOST11** 起各位依次被设置为 **1**，并保持 **1**。

2. 振荡稳定时间计数器的计数达到 **OSTS** 设置的振荡稳定时间。如果已进入 **STOP** 模式然后释放，并使用内部高速振荡时钟作为 **CPU** 时钟时，则设置振荡稳定时间如下。

- 预期的 **OSTC** 振荡稳定时间 ≤ 由 **OSTS** 设置的振荡稳定时间

因此需要注意，在释放 **STOP** 模式后，只有经过 **OSTS** 设置的振荡稳定时间，对应状态才会被设置到 **OSTC**。

3. X1 时钟振荡稳定等待时间不包括从 **STOP** 模式释放到时钟振荡启动的这段时间（下图的**"a"**）。



备注 f_X： X1 时钟振荡频率

（2）　振荡稳定时间选择寄存器（OSTS）

该寄存器用于选择释放 STOP 模式后 X1 时钟振荡稳定等待时间。

当 X1 时钟被选择作为 CPU 时钟，在释放 STOP 模式后，操作等待由 OSTS 设置的时间。

当内部高速振荡时钟被选择作为 CPU 时钟，在释放 STOP 模式后，通过 OSTC 来确认期望的振荡稳定时间已经经过。可以检查振荡稳定时间，直到经过 OSTC 设置的时间。

可以通过 8 位存储器操作指令来设置 OSTS。

复位信号的产生会将 OSTS 设置为 05H。

图 22-2. 振荡稳定时间选择寄存器（OSTS）的格式

地址：　FFA4H　复位后：　05H　R/W

符号	7	6	5	4	3	2	1	0
OSTS	0	0	0	0	0	OSTS2	OSTS1	OSTS0

OSTS2	OSTS1	OSTS0	振荡稳定时间的选择	
				f_X = 10 MHz
0	0	1	$2^{11}/f_X$	204.8 μs
0	1	0	$2^{13}/f_X$	819.2 μs
0	1	1	$2^{14}/f_X$	1.64 ms
1	0	0	$2^{15}/f_X$	3.27 ms
1	0	1	$2^{16}/f_X$	6.55 ms
其它情况			禁止设置	

注意事项

1. **要在 X1 时钟用作 CPU 时钟时设置 STOP 模式，则应在执行 STOP 指令之前设置 OSTS。**

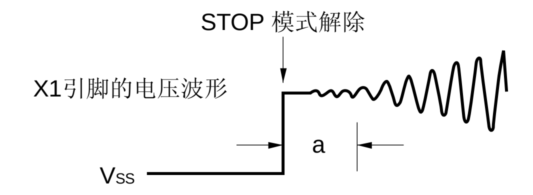
2. **在 X1 时钟振荡稳定期间，不要改变 OSTS 寄存器的值。**

3. **振荡稳定时间计数器的计数达到 OSTS 设置的振荡稳定时间。如果已进入 STOP 模式然后释放，并使用内部高速振荡时钟作为 CPU 时钟时，则设置振荡稳定时间如下。**

- **预期的 OSTC 振荡稳定时间 ≤ 由 OSTS 设置的振荡稳定时间**

因此需要注意，在释放 STOP 模式后，只有经过 OSTS 设置的振荡稳定时间，对应状态才会被设置到 OSTC。

4. **X1 时钟振荡稳定等待时间不包括从 STOP 模式释放到时钟振荡启动的这段时间（下图的“a”）。**



备注　f_X：X1 时钟振荡频率。

22.2 待机功能的操作

22.2.1 HALT模式

（1）　HALT 模式

通过执行 HALT 指令设置 HALT 模式。无论设置之前 CPU 使用高速系统时钟、内部高速振荡时钟还是副系统时钟，都可以设置 HALT 模式。

HALT 模式下的操作状态如下所示。

表 22-1. HALT 模式中的操作状态（1/2）

HALT 模式设置 / 项目			当 CPU 运行于主系统时钟，执行 HALT 指令		
			当 CPU 运行于内部高速振荡时钟（f_{RH}）时	当 CPU 运行于 X1 时钟（f_X）时	当 CPU 运行于外部主系统时钟（f_{EXCLK}）时
系统时钟			供给 CPU 的时钟被停止		
	主系统时钟	f_{RH}	操作继续（不能停止）	保持设置 HALT 模式前的状态	
		f_X	保持设置 HALT 模式前的状态	操作继续（不能停止）	保持设置 HALT 模式前的状态
		f_{EXCLK}	由外部时钟输入操作或停止		操作继续（不能停止）
	副系统时钟	f_{XT}	保持设置 HALT 模式前的状态		
	f_{RL}		保持设置 HALT 模式前的状态		
CPU			操作停止		
Flash 存储器			操作停止		
RAM			保持设置 HALT 模式前的状态		
端口（锁存器）			保持设置 HALT 模式前的状态		
16 位定时器/事件计数器 00			可操作的		
8 位定时器/事件计数器		50			
		51			
		52			
8 位定时器		H0			
		H1			
		H2			
实时计数器					
看门狗定时器			可操作。当通过选项字节设置为“可由软件停止内部低速振荡器”时，停止看门狗定时器的时钟。		
蜂鸣器输出			可操作的		
10 位逐次逼近型 A/D 转换器					
16 位ΔΣ型 A/D 转换器					
串行接口		UART0			
		UART6			
		CSI10			
LCD 控制器/驱动器					
曼彻斯特码发生器					
遥控接收器					
上电清零（POC）功能					
低电压检测功能					
外部中断					

备注　f_{RH}：　内部高速振荡时钟

f_X：　X1 时钟

f_{EXCLK}：　外部主系统时钟

f_{XT}：　XT1 时钟

f_{RL}：　内部低速振荡时钟

表 22-1. HALT 模式中的操作状态（2/2）

项目 \ HALT 模式设置			当 CPU 运行于副系统时钟，执行 HALT 指令
			当 CPU 运行于 XT1 时钟（f_{XT}）时
系统时钟			供给 CPU 的时钟被停止
	主系统时钟	f_{RH}	保持设置 HALT 模式前的状态
		f_X	
		f_{EXCLK}	由外部时钟输入操作或停止
	副系统时钟	f_{XT}	操作继续（不能停止）
	f_{RL}		保持设置 HALT 模式前的状态
CPU			操作停止
Flash 存储器			操作停止
RAM			保持设置 HALT 模式前的状态
端口（锁存器）			保持设置 HALT 模式前的状态
16 位定时器/事件计数器 00[注1]			可操作的
8 位定时器/事件计数器	50[注1]		
	51[注1]		
	52[注1]		
8 位定时器	H0		
	H1		
	H2		
实时计数器			
看门狗定时器			可操作。当通过选项字节设置为“可由软件停止内部低速振荡器”时，停止看门狗定时器的时钟。
蜂鸣器输出			可操作的。但是，当外设硬件时钟停止时（f_{PRS}），禁止操作。
10 位逐次逼近型 A/D 转换器			
16 位 ΔΣ 类型 A/D 转换器			可操作的
实时计数器			
串行接口	UART0		
	UART6		
	CSI10[注]		
LCD 控制器/驱动器			
曼彻斯特码发生器			
遥控器接收器			
上电清零（POC）功能			
低电压检测功能			
外部中断			

注　当 CPU 运行于副系统时钟且内部高速振荡时钟已经停止时，不要启动使用外部时钟的功能，外部时钟从外设硬件引脚输入。

备注　f_{RH}：　内部高速振荡时钟
f_X：　X1 时钟
f_{EXCLK}：　外部主系统时钟
f_{XT}：　XT1 时钟
f_{RL}：　内部低速振荡时钟

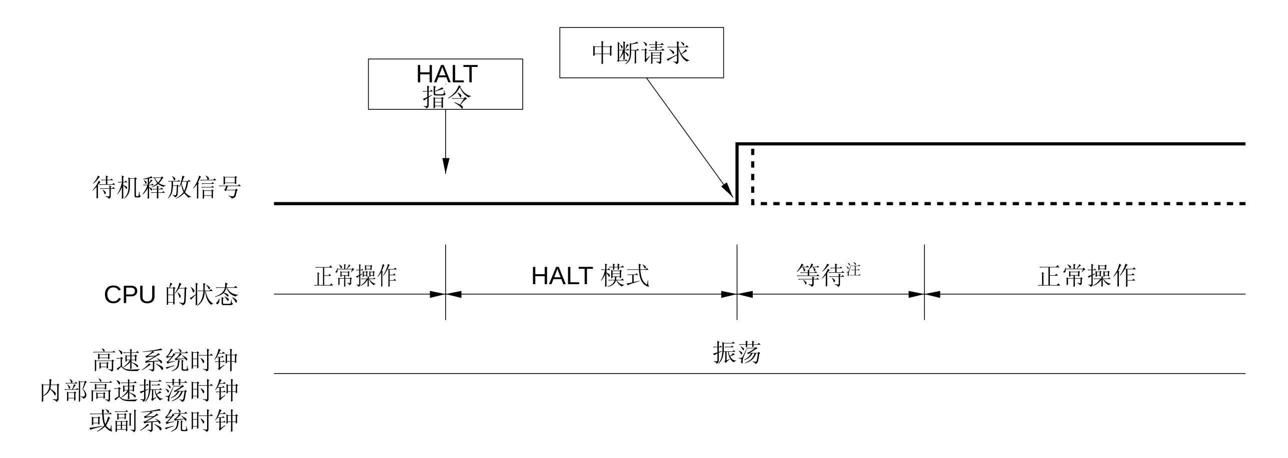
（2）　释放 HALT 模式

通过下列两个源可以释放 HALT 模式。

（a）由未屏蔽的中断请求释放

当产生一个未屏蔽的中断时，HALT 模式被释放。如果使能中断响应，则执行向量中断服务。如果禁止中断响应，则执行下一个地址的指令。

图 22-3. 通过产生中断请求释放 HALT 模式



注　等待时间如下：

- 执行向量中断服务时：　8 或 9 个时钟
- 不执行向量中断服务时：　2 或 3 个时钟

备注　虚线表示释放待机模式的中断请求被响应的情况。

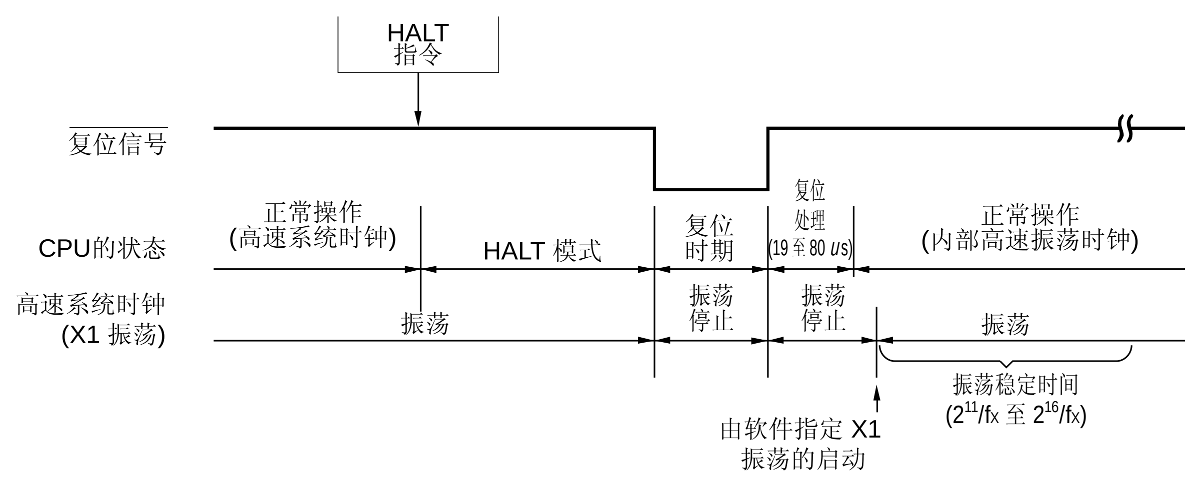
（b） 由产生的复位信号释放

当产生复位信号时，释放 HALT 模式，然后在正常复位操作的情况下，跳转到复位矢量地址后，开始执行程序。

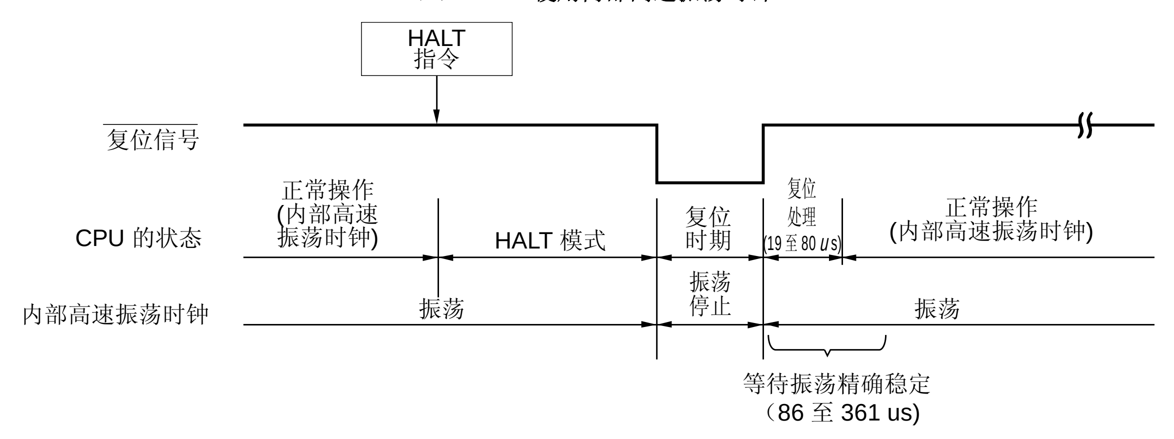
<R>

图 22-4. 通过复位释放 HALT 模式

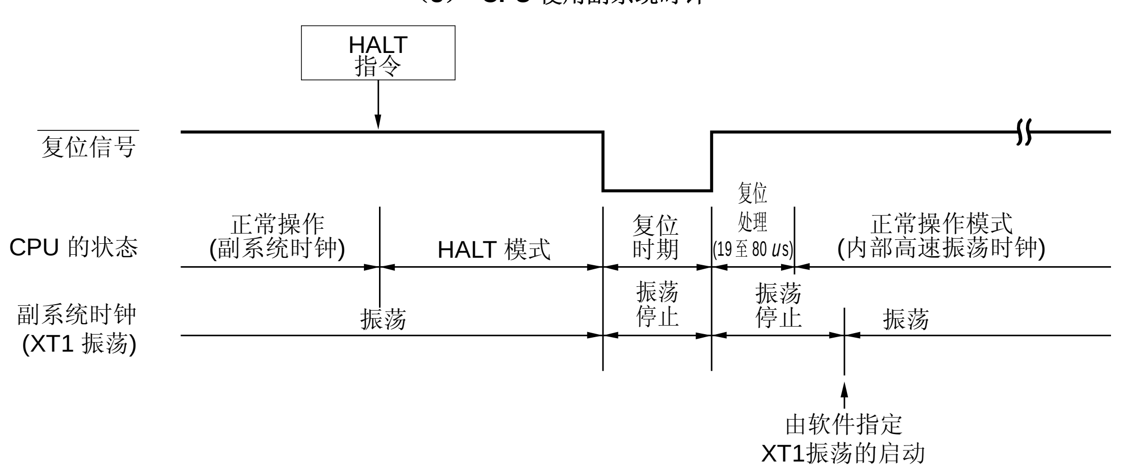
（1） CPU 使用高速系统时钟



（2） CPU 使用内部高速振荡时钟



（3） CPU 使用副系统时钟



备注　　fx：　X1 时钟振荡频率

表 **22-2. HALT** 模式下中断请求对应的操作

<table>
<tr><th>释放源</th><th>MK××</th><th>PR××</th><th>IE</th><th>ISP</th><th>操作</th></tr>
<tr><td rowspan="6">可屏蔽中断请求</td><td>0</td><td>0</td><td>0</td><td>×</td><td>执行下一个地址的指令</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>×</td><td>执行中断服务</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td rowspan="2">执行下一个地址的指令</td></tr>
<tr><td>0</td><td>1</td><td>×</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>执行中断服务</td></tr>
<tr><td>1</td><td>×</td><td>×</td><td>×</td><td>保持 HALT 模式</td></tr>
<tr><td>复位</td><td>–</td><td>–</td><td>×</td><td>×</td><td>复位处理</td></tr>
</table>

×：无须理会

22.2.2 STOP 模式

（1） STOP 模式设置及操作状态

通过执行 STOP 指令设置 STOP 模式。只有当设置 STOP 模式之前 CPU 使用主系统时钟时，才可以设置 STOP 模式。

注意事项　**因为中断请求信号用来释放待机模式，因此如果一个中断源的中断请求标志被置位且中断屏蔽标志被重置时，如果该中断源置位则立即释放待机模式。于是，在执行 STOP 指令后，立即将 STOP 模式重置为 HALT 模式，并且在经过 OSTS 设置的等待时间后，系统马上返回操作模式。**

STOP 模式下的操作状态如下所示。

表 22-3. STOP 模式下的操作状态

项目 ＼ STOP 模式设置			CPU 运行于主系统时钟，执行 STOP 指令		
			当 CPU 运行于内部高速振荡时钟（f_{RH}）时	当 CPU 运行于 X1 时钟（f_X）时	当 CPU 运行于外部主系统时钟（f_{EXCLK}）时
系统时钟			供给 CPU 的时钟被停止		
	主系统时钟	f_{RH}	停止		
		f_X			
		f_{EXCLK}	输入无效		
	副系统时钟	f_{XT}	保持设置 STOP 模式前的状态		
	f_{RL}		保持设置 STOP 模式前的状态		
CPU			操作停止		
Flash 存储器			操作停止		
RAM			保持设置 STOP 模式前的状态		
端口（锁存器）			保持设置 STOP 模式前的状态		
16 位定时器/时间计数器 00[注1]			仅当选择 TM52 输出或 TI000 作为计数时钟时可以操作		
8 位定时器/事件计数器	50[注1]		仅当选择 TI50 作为计数时钟时可以操作		
	51[注1]		仅当选择 TI51 作为计数时钟时可以操作		
	52[注1]		仅当选择 TI52 作为计数时钟时可以操作		
8 位定时器	H0		仅当 8 位定时器/事件计数器 50 操作期间，选择 TM50 输出作为计数时钟时可以操作		
	H1		仅当选择 f_{RL}，$f_{RL}/2^7$，$f_{RL}/2^9$ 作为计数时钟时可以操作		
	H2		操作停止		
实时计数器			仅当选择副系统时钟作为计数时钟时可以操作		
看门狗定时器			可操作的。当选项字节设置为“可由软件停止内部低速振荡器”时，停止看门狗定时器的时钟。		
蜂鸣器输出			操作停止		
10 位逐次逼近型 A/D 转换器					
<R> 16 位 ΔΣ 型 A/D 转换器[注2]			可操作的 Operable		
串行接口	UART0		仅当 8 位定时器/事件计数器 50 操作期间，选择 TM50 输出作为计数时钟时可以操作		
	UART6				
	CSI10[注1]		仅当选择外部时钟作为串行时钟时可以操作		
LCD 控制器/驱动器			仅当选择副系统时钟作为计数时钟时可以操作		
曼彻斯特码发生器			操作停止		
遥控器接收器			仅当选择副系统时钟作为计数时钟时可以操作		
上电清零（POC）功能			可操作的		
低电压检测功能					
外部中断					

注　**1.** 在 STOP 模式下，不要启动使用外部时钟的功能，外部时钟从外设硬件引脚输入。

<R>　**2.** 当执行 STOP 指令时，如果选择 $f_{PRS}/4$，$f_{PRS}/8$ 或 $f_{PRS}/16$ 作为采样时钟，请确保关闭 16 位 ΔΣ 型 A/D 转换器的电源（ADDPON = 0）。

备注　f_{RH}：　内部高速振荡时钟
　　f_X：　X1 时钟
　　f_{EXCLK}：　外部主系统时钟
　　f_{XT}：　XT1 时钟
　　f_{RL}：　内部低速振荡时钟

注意事项 1. 在 **STOP** 模式释放后，要使用在 **STOP** 模式下停止操作的外设硬件，以及在 **STOP** 模式下时钟停止振荡的外设硬件，必须重新启动外设硬件。

2. 即使已经通过选项字节选择“可由软件停止内部低速振荡器”，在 **STOP** 模式下，内部低速振荡时钟仍保持 **STOP** 模式设置前的状态继续振荡。要停止内部低速振荡器在 **STOP** 模式下的振荡，应通过软件停止该振荡器，然后执行 **STOP** 指令。

<R> 3. 当 **CPU** 使用高速系统时钟（**X1** 振荡）时，要缩短 **STOP** 模式释放后的振荡稳定时间，可按照下列过程，在执行 **STOP** 指令之前将 **CPU** 时钟切换到内部高速振荡时钟。

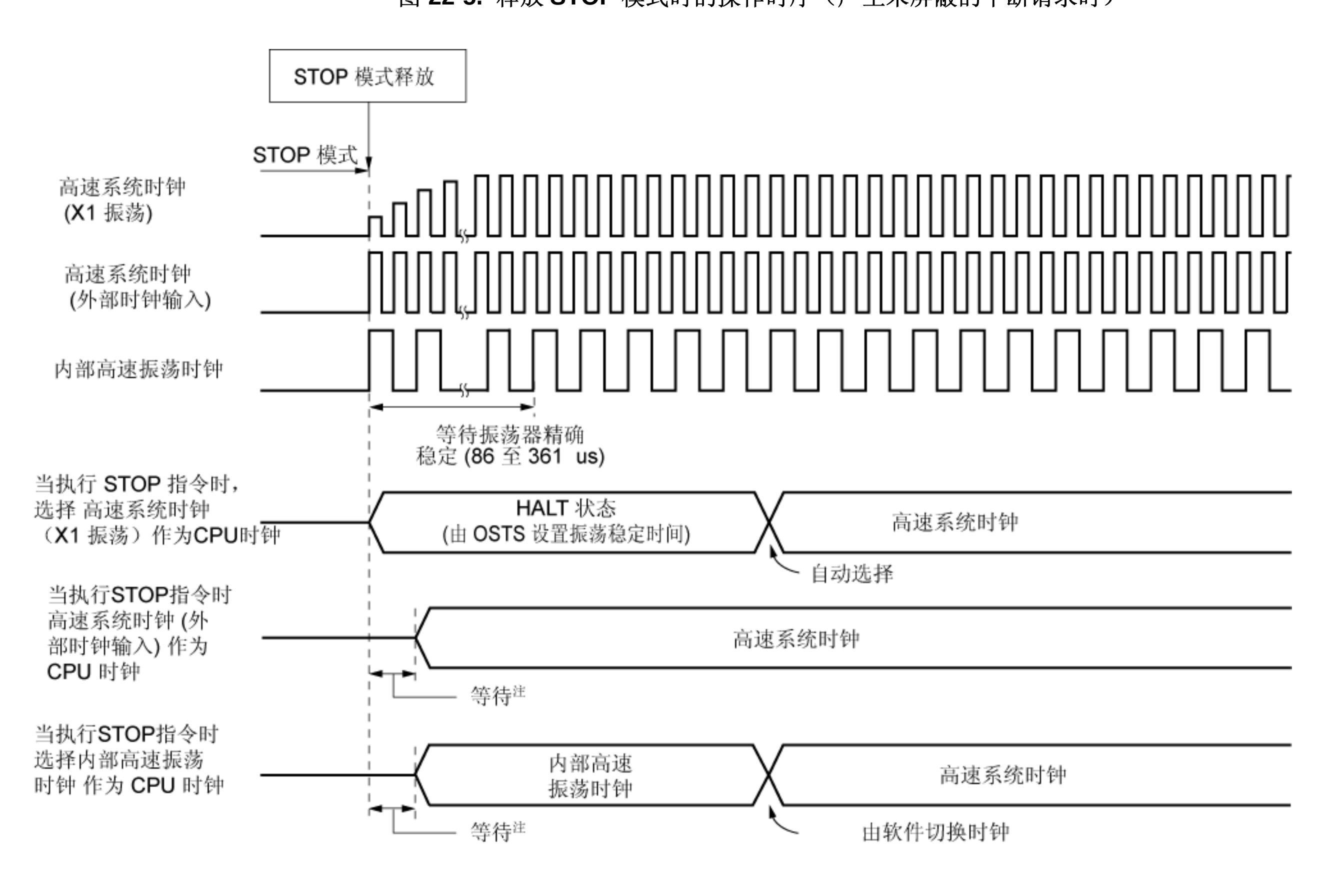
<1> 设置 **RSTOP** 为 **0**（启动内部高速振荡器的振荡）→ **<2>** 设置 **MCM0** 为 **0**（将 **CPU** 从 **X1** 振荡切换到内部高速振荡）→ **<3>** 检查 **MCS** 为 **0**（检查 **CPU** 时钟）→ **<4>** 检查 **RSTS** 为 **1**（检查内部高速振荡操作）→ **<5>** 执行 **STOP** 指令。

在 **STOP** 模式释放后，在将 **CPU** 时钟从内部高速振荡时钟切换到高速系统时钟（**X1** 振荡）之前，应使用振荡稳定时间计数器的状态寄存器（**OSTC**）来检查振荡稳定时间。

<R> 4. 在已经确认内部高速振荡器稳定工作（**RSTS = 1**）之后，执行 **STOP** 指令。

（**2**）　**STOP** 模式的释放

图 **22-5.** 释放 **STOP** 模式时的操作时序（产生未屏蔽的中断请求时）



注　　等待时间如下所示：

- 执行向量中断服务时：　8 或 9 个时钟
- 不执行向量中断服务时：　2 或 3 个时钟

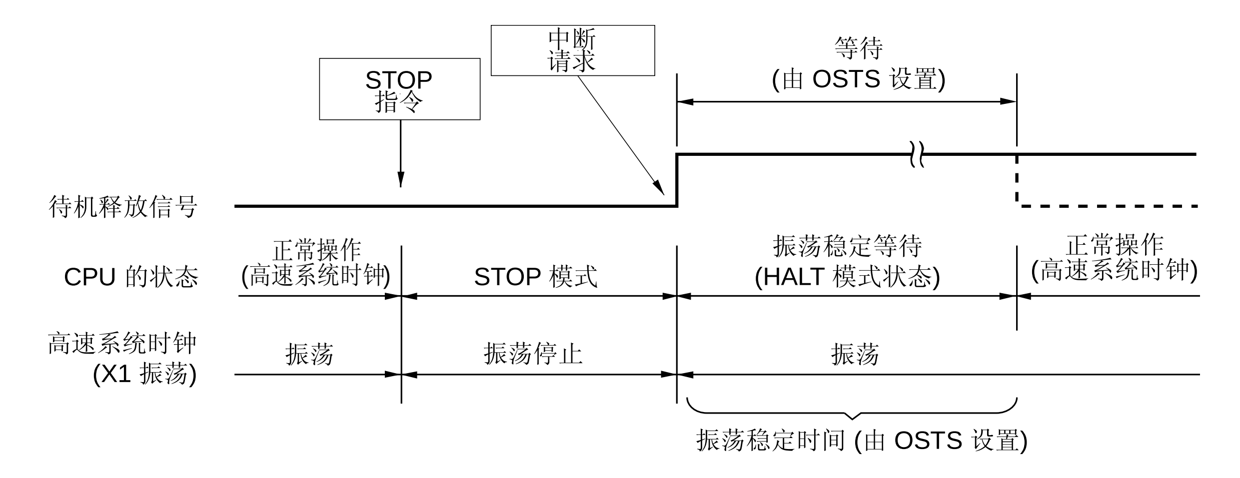
通过下列两个源可以释放 STOP 模式。

（a） 由未屏蔽的中断请求释放

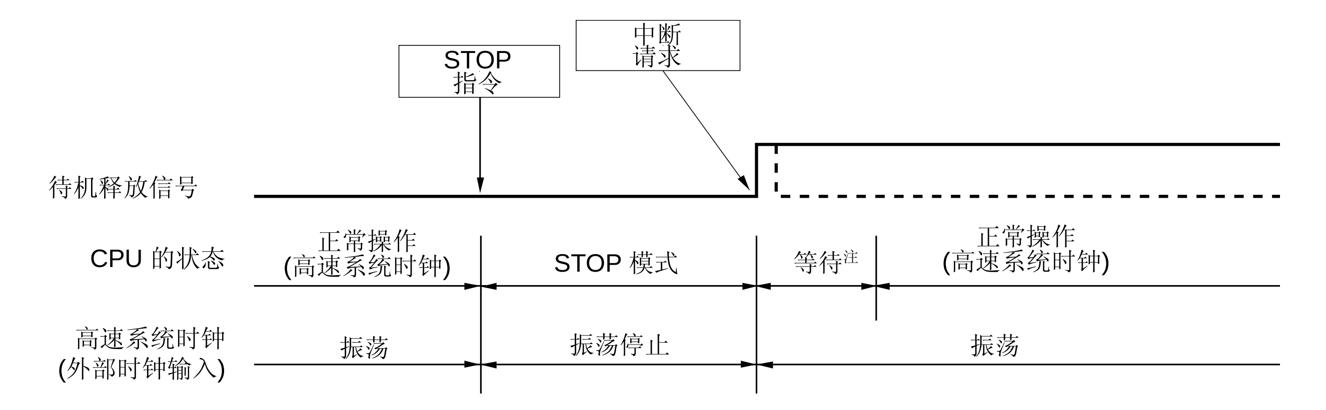
当产生一个未屏蔽的中断时，STOP 模式被释放。经历振荡稳定时间后，如果使能中断响应，则执行向量中断服务。如果禁止中断响应，则执行下一个地址的指令。

图 22-6. 通过产生中断请求释放 STOP 模式（1/2）

（1） 当 CPU 使用高速系统时钟（X1 振荡器）



（2） 当 CPU 使用高速系统时钟（外部时钟输入）



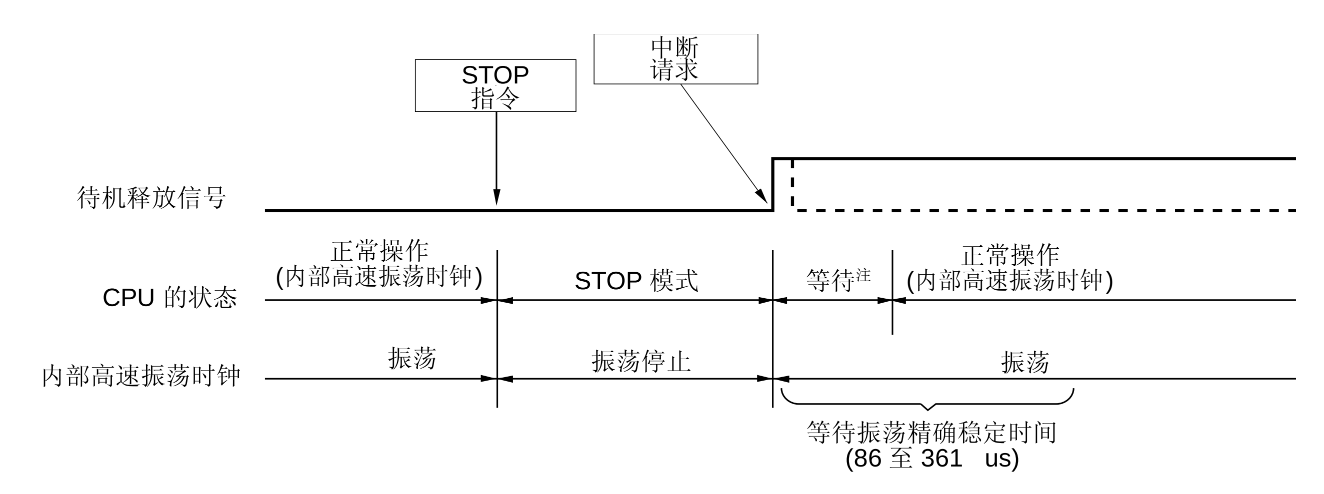
注　　等待时间如下所示：

- 执行向量中断服务时：　　8 或 9 个 时钟
- 不执行向量中断服务时：　2 或 3 个 时钟

备注　　虚线表示释放待机模式的中断请求被响应的情况。

图 22-6. 通过产生中断请求释放 STOP 模式（2/2）

（3）　当 CPU 使用内部高速振荡时钟



注　等待时间如下所示：

- 执行向量中断服务程序时：　8 或 9 个时钟
- 不执行向量中断服务程序时：　2 或 3 个时钟

备注　虚线表示释放待机模式的中断请求被响应的情况。

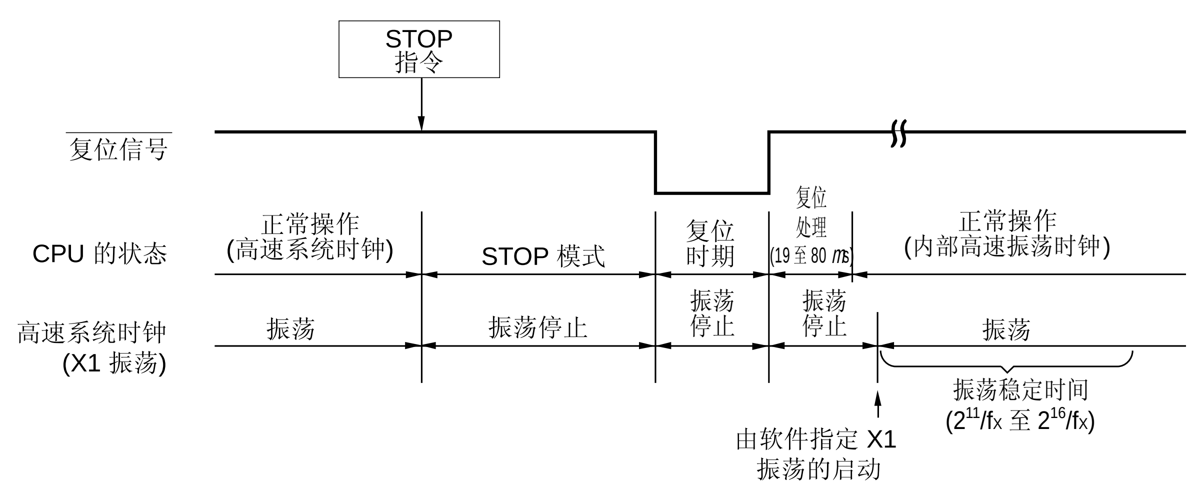
（**b**） 由产生的复位信号释放

当产生复位信号时，释放 STOP 模式，然后在正常复位操作的情况下，跳转到复位矢量地址后，开始执行程序。

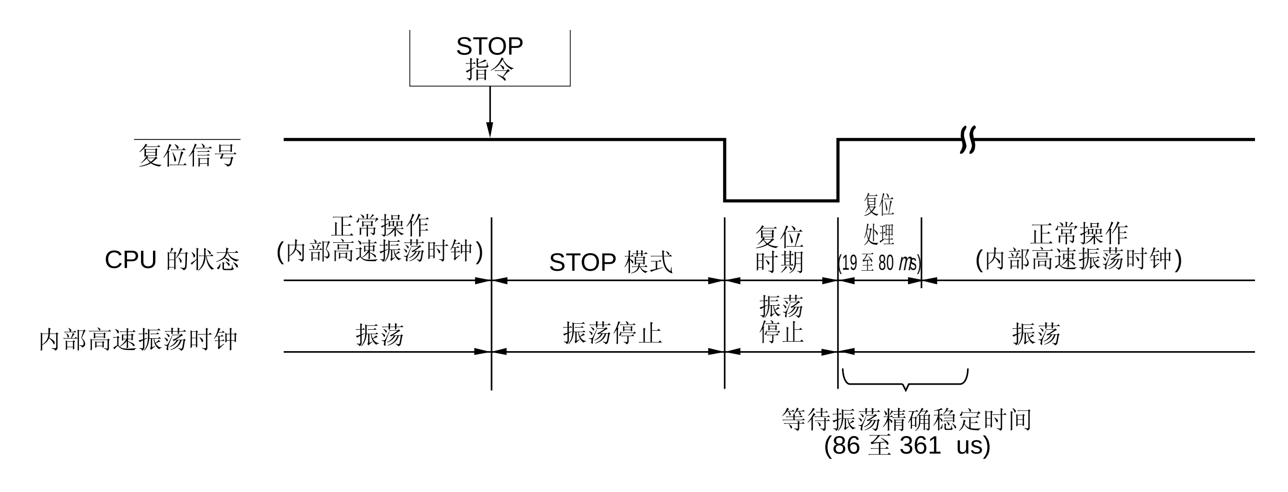
<R>

图 22-7. 由复位释放 STOP 模式

（1） 当 CPU 使用高速系统时钟



（2） 当 CPU 使用内部高速振荡时钟



备注　fx： X1 时钟振荡频率

表 22-4. STOP 模式下中断请求对应的操作

释放源	MK××	PR××	IE	ISP	操作
可屏蔽的中断请求	0	0	0	×	执行下一个地址的指令
	0	0	1	×	执行中断服务
	0	1	0	1	执行下一个地址的指令
	0	1	×	0	
	0	1	1	1	执行中断服务
	1	×	×	×	保持 STOP 模式
复位	–	–	×	×	复位处理

×： 无须理会

第二十三章　复位功能

以下 4 种操作用于产生复位信号。

（1）由 $\overline{\text{RESET}}$ 引脚输入的外部复位信号
（2）由看门狗定时器程序循环检测引起的内部复位
（3）通过比较供电电压和上电清零（POC）电路的检测电压引起的内部复位
（4）通过比较供电电压和低电压检测电路（LVI）的检测电压引起的内部复位

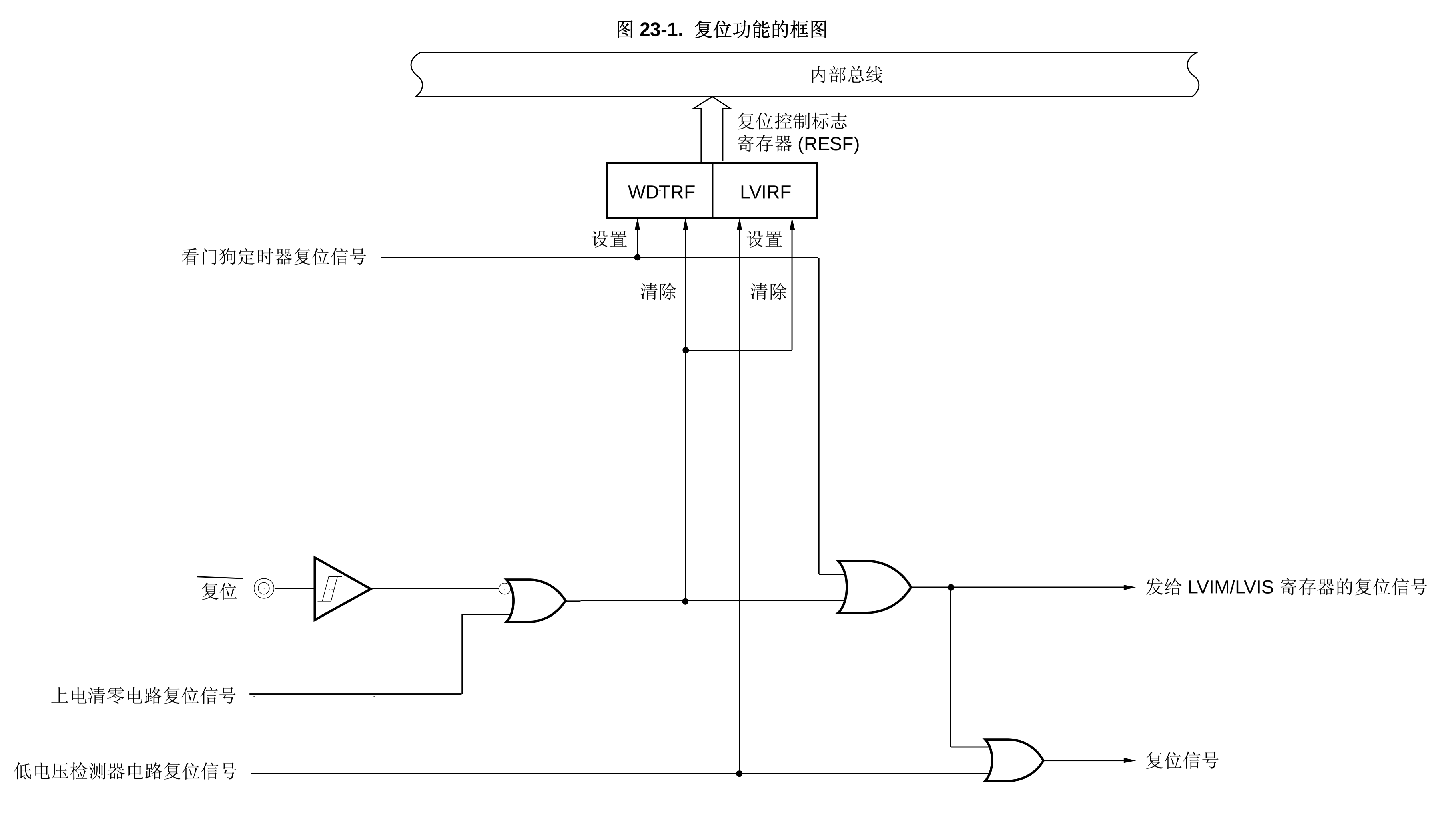
外部复位与内部复位在功能上没有区别。在这两种情况中，当产生复位信号时，程序都是从地址 0000H 和 0001H 开始执行。

当低电平输入到 $\overline{\text{RESET}}$ 引脚、看门狗定时器溢出或 POC 和 LVI 电路的电压检测都可以引起复位，每项硬件的设置状态如表 23-1 和 23-2 所示。在复位信号产生期间或在复位释放后的振荡稳定时间内，每个引脚均为高阻抗。

当低电平输入到 $\overline{\text{RESET}}$ 引脚时，设备被复位。当 $\overline{\text{RESET}}$ 引脚输入高电平时，设备从复位状态释放，并在复位处理后使用内部高速振荡时钟执行程序。由看门狗定时器引起的复位自动释放，并在复位处理后使用内部高速振荡时钟执行程序（参见**图 23-2** 至 **23-4**）。由 POC 和 LVI 电路供电电压检测引起的复位，在复位后，当 $V_{DD} \geq V_{POC}$ 或 $V_{DD} \geq V_{LVI}$ 时自动释放，并使用内部高速振荡时钟执行程序（参见**第二十四章 上电清零电路** 和 **第二十四章 低电压检测电路**）。

注意事项　1. 对于外部复位，输入 $\overline{\text{RESET}}$ 引脚的低电平的至少为 10 μs。
2. 复位输入期间，X1 时钟、XT1 时钟、内部高速振荡时钟和内部低速振荡时钟停止振荡。外部主系统时钟输入无效。
3. 在通过复位释放 STOP 模式时，复位输入期间保持 STOP 模式的内容。但端口引脚变为高阻态。

图 23-1. 复位功能的框图

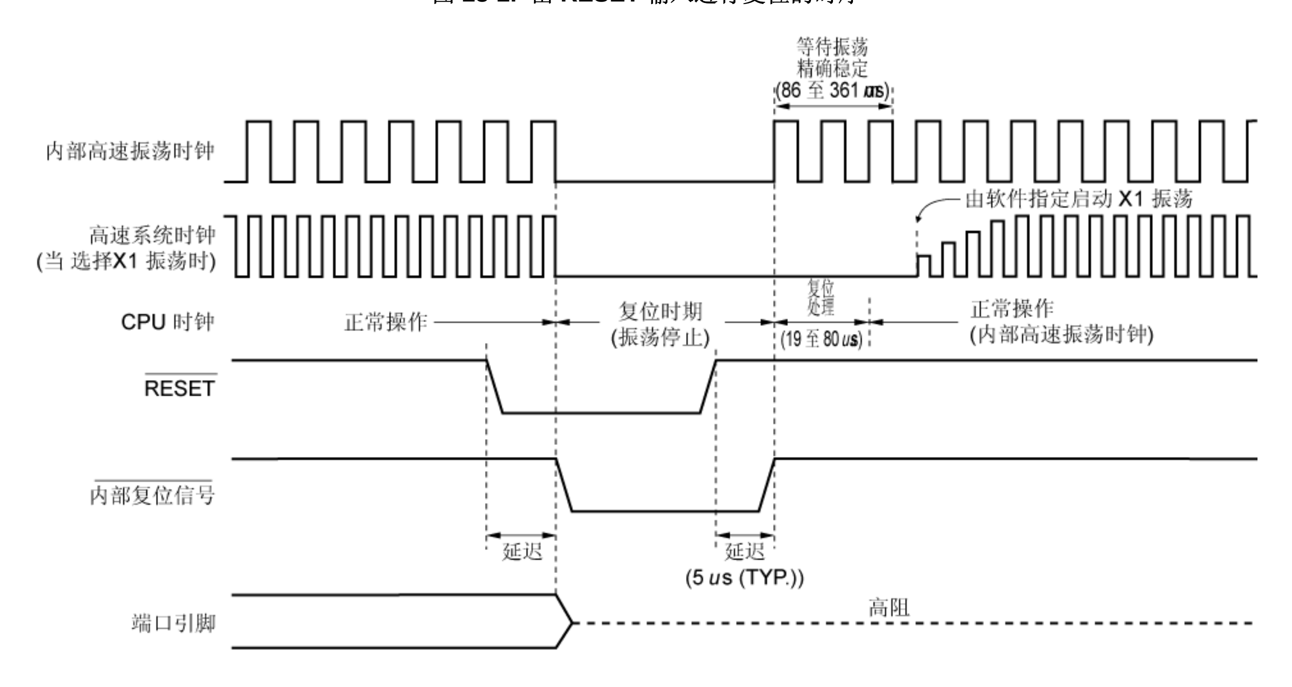


注意事项　LVI 电路内部复位不能复位 LVI 电路。

备注　**1.** LVIM：　低电压检测寄存器
　　　　2. LVIS：　低电压检测电平选择寄存器

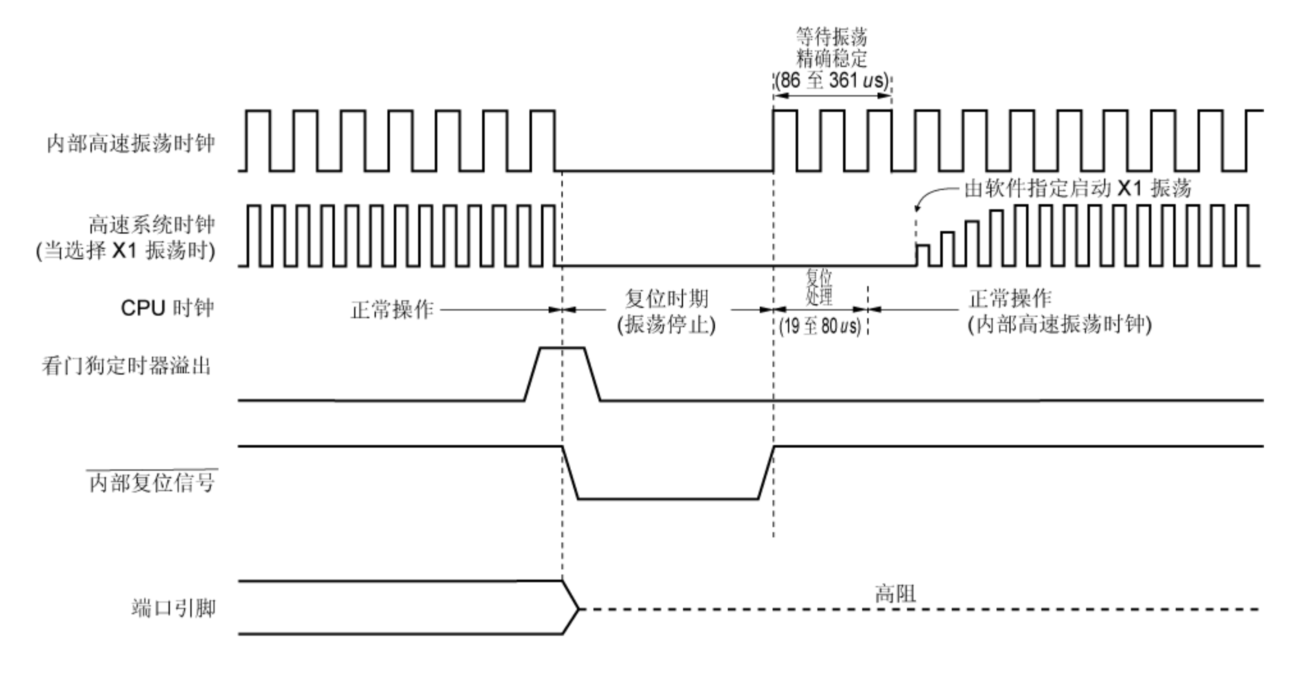
<R>

图 23-2. 由 RESET 输入进行复位的时序



<R>

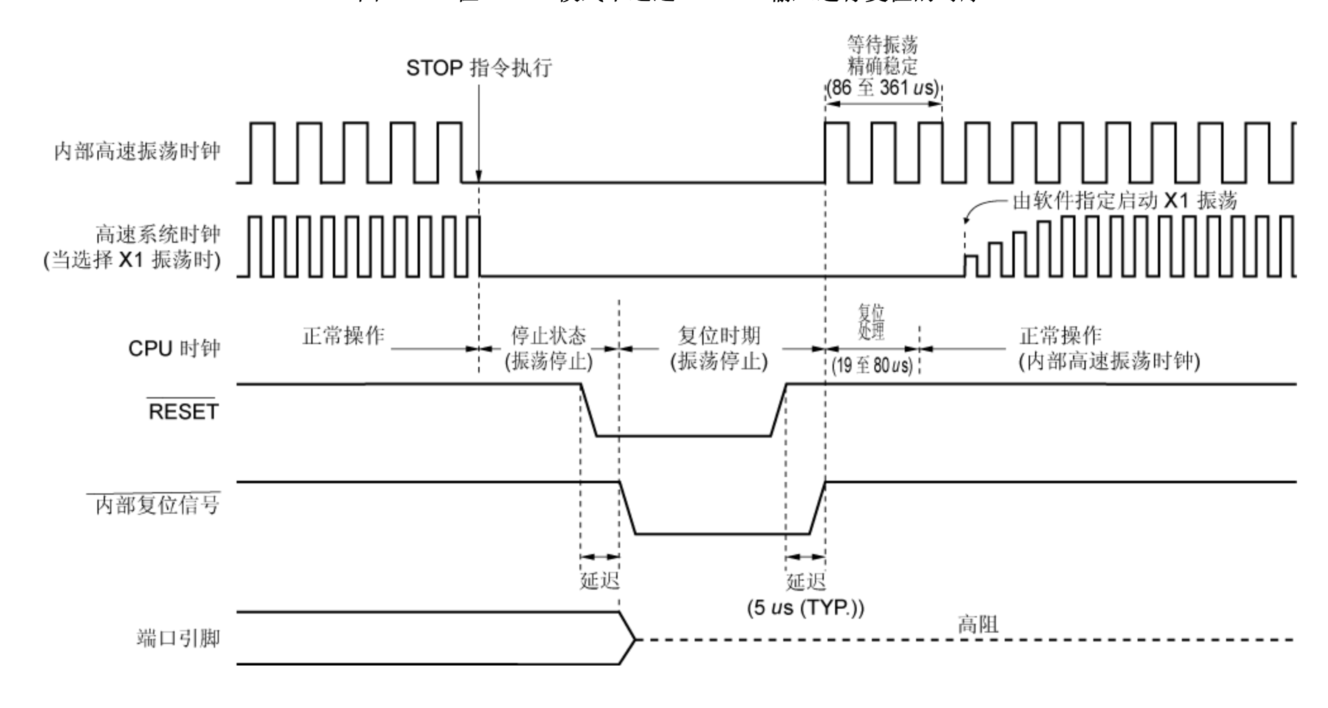
图 23-3. 由于看门狗定时器溢出进行复位的时序



注意事项　看门狗定时器内部复位功能也会复位看门狗定时器。

<R>

图 23-4. 在 STOP 模式下通过 $\overline{\text{RESET}}$ 输入进行复位的时序



备注　上电清零（POC）电路和低电压检测电路的复位时序，参见**第二十四章　上电清零电路** 和 **第二十五章　低电压检测电路**。

表 23-1. 复位期间的操作状态

项目			复位期间
系统时钟			供给 CPU 的时钟停止
	主系统时钟	f_{RH}	操作停止
		f_X	操作停止（引脚为 I/O 端口模式）
		f_{EXCLK}	时钟输入无效 （引脚为 I/O 端口模式）
	副系统时钟	f_{XT}	操作停止 （引脚为 I/O 端口模式）
	f_{RL}		操作停止
CPU			
Flash 存储器			
RAM			
端口 （锁存器）			
16 位定时器/事件计数器 00			
8 位定时器/事件计数器		50	
		51	
		52	
8 位定时器		H0	
		H1	
		H2	
实时计数器			
看门狗定时器			
蜂鸣器输出			
10 位逐次逼近型 A/D 转换器			
16 位ΔΣ 型 A/D 转换器			
串行接口		UART0	
		UART6	
		CSI10	
LCD 控制/驱动器			
曼彻斯特码发生器			
遥控器接收器			
上电清零功能			可操作的
低电压检测功能			操作停止
外部中断			

备注
f_{RH}：　内部高速振荡时钟
f_X：　X1 振荡时钟
f_{EXCLK}：　外部主系统时钟
f_{XT}：　XT1 振荡时钟
f_{RL}：　内部低速振荡时钟

表 23-2. 复位响应后的硬件状态（1/4）

硬件		响应复位后[注1]
程序计数器 （PC）		设置为复位向量表（0000H，0001H）的内容
堆栈指针 （SP）		不确定
程序状态字 （PSW）		02H
RAM	数据存储器	不确定[注2]
	通用寄存器	不确定[注2]
端口寄存器 （P1 至 P4，P8，P10 至 P12，P14，P15） （输出锁存）		00H
端口模式寄存器 （PM1 至 PM4，PM8，PM10 至 PM12，PM14，PM15）		FFH
上拉电阻选项寄存器 （PU1，PU3，PU4，PU8，PU10 至 PU12，PU14，PU15）		00H
端口功能寄存器 （PF1）		00H
端口功能寄存器（PF2）		00H
端口功能寄存器（PFALL）		00H
内部扩展 RAM 容量切换寄存器（IXS）		0CH[注3]
内部存储器容量切换寄存器（IMS）		CFH[注3]
时钟输出模式选择寄存器（OSCCTL）		00H
处理器时钟控制寄存器 （PCC）		01H
内部振荡模式寄存器（RCM）		80H
主 OSC 控制寄存器（MOC）		80H
主时钟模式寄存器 （MCM）		00H
振荡稳定时间计数器状态寄存器（OSTC）		00H
振荡稳定时间选择寄存器 （OSTS）		05H
内部高速振荡修正寄存器（HIOTRM）		10H
16 位定时器/事件计数器 00	定时器计数器 00 （TM00）	0000H
	捕获/比较寄存器 000，010 （CR000，CR010）	0000H
	模式控制寄存器 00 （TMC00）	00H
	预分频器模式寄存器 00 （PRM00）	00H
	捕获/比较控制寄存器 00 （CRC00）	00H
	定时器输出控制寄存器 00 （TOC00）	00H
8 位定时器/事件计数器 50，51，52	定时器计数器 50，51，52 （TM50，TM51，TM52）	00H
	比较寄存器 50，51，52 （CR50，CR51，CR52）	00H
	定时器时钟选择寄存器 50，51，52 （TCL50，TCL51，TCL52）	00H
	模式控制寄存器 50，51 ，52（TMC50，TMC51，TMC52）	00H

注
1. 在产生复位信号期间或振荡稳定时间等待期间，硬件状态中只有 PC 内容不确定。复位后其它硬件状态保持不变。
2. 当在待机模式下执行复位时，即使复位后，也保持复位前的状态。
3. 无论内部存储器容量大小，复位释放后 78K0/LE3 所有产品的内部存储器容量切换寄存器（IMS）和内部扩展 RAM 容量切换寄存器（IXS）的初始值都是固定的（IMS = CFH，IXS = 0CH）。因此，复位释放后请确保对各个产品设置下面的值。

Flash 存储器版本（78K0/LE3）	IMS	IXS
μPD78F0441，78F0451，78F0461	04H	0CH
μPD78F0442，78F0452，78F0462	C6H	
μPD78F0443，78F0453，78F0463	C8H	
μPD78F0444，78F0454，78F0464	CCH	0AH
μPD78F0445，78F0455，78F0465	CFH	

表 23-2. 复位响应后的硬件状态 （2/4）

硬件		响应复位后的状态[注 1]
8 位计数器 H0，H1，H2	比较寄存器 00，10，01，11，02，12（CMP00，CMP10，CMP01，CMP11，CMP02，CMP12）	00H
	模式寄存器 （TMHMD0，TMHMD1，TMHMD2）	00H
	载波控制寄存器 1（TMCYC1）[注 2]	00H
实时计数器	时钟选择寄存器（RTCCL）	00H
	副计数寄存器（RSUBC）	0000H
	秒计数寄存器 （SEC）	00H
	分技术寄存器 （MIN）	00H
	小时技术寄存器 （HOUR）	12H
	周计数寄存器 （WEEK）	00H
	天计数寄存器 （DAY）	01H
	月计数寄存器 （MONTH）	01H
	年计数寄存器 （YEAR）	00H
	钟表误差修正计数器 （SUBCUD）	00H
	闹钟分钟寄存器 （ALARMWM）	00H
	闹钟小时寄存器 （ALARMWH）	12H
	闹钟星期寄存器 （ALARMWW）	00H
	控制寄存器 0 （RTCC0）	00H
	控制寄存器 1 （RTCC1）	00H
	控制寄存器 2 （RTCC2）	00H
蜂鸣器输出控制器	时钟输出选择寄存器 （CKS）	00H
看门狗定时器	使能寄存器 （WDTE）	1AH/9AH [注 3]
10 位逐次逼近型 A/D 转换器	10 位 A/D 转换结果寄存器 （ADCR）	0000H
	8 位 A/D 转换结果寄存器（ADCRH）	00H
	模式寄存器 （ADM）	00H
	模拟输入通道选择寄存器（ADS）	00H
	A/D 端口配置寄存器 （ADPC0）	08H
16 位ΔΣ 型 A/D 转换器	ΔΣ A/D 转换器控制寄存器 0 （ADDCTL0）	00H
	ΔΣ A/D 转换器控制寄存器 1 （ADDCTL1）	00H
	16 位ΔΣ A/D 转换状态寄存器（ADDSTR）	00H
	16 位ΔΣ A/D 转换结果寄存器（ADDCR）	0000H
	8 位ΔΣ A/D 转换结果寄存器（ADDCRH）	00H
串行接口 UART0	接收缓冲寄存器 0 （RXB0）	FFH
	发送移位寄存器 0 （TXS0）	FFH
	异步串行接口操作模式寄存器 0 （ASIM0）	01H
	异步串行接口接收错误状态寄存器 0 （ASIS0）	00H
	波特率发生器控制寄存器 0 （BRGC0）	1FH

注 **1.** 在产生复位信号期间或振荡稳定时间等待期间，硬件状态中只有 PC 内容不确定。复位后其它硬件状态保持不变。

2. 仅限 8 位定时器 H1。

3. 通过设置选项字节决定 WDTE 的复位值。

表 23-2. 复位响应后的硬件状态（3/4）

硬件		响应复位后的状态[注 1]
串行接口 UART6	接收缓冲寄存器 6 （RXB6）	FFH
	发送缓冲寄存器 6 （TXB6）	FFH
	异步串行接口操作模式寄存器 6 （ASIM6）	01H
	异步串行接口接收错误状态寄存器 6 （ASIS6）	00H
	异步串行接口发送状态寄存器 6 （ASIF6）	00H
	时钟选择寄存器 6 （CKSR6）	00H
	波特率发生器控制寄存器 6 （BRGC6）	FFH
	异步串行接口控制寄存器 6 （ASICL6）	16H
	输入切换控制寄存器 （ISC）	00H
串行接口 CSI10	发送缓冲器寄存器 10 （SOTB10）	00H
	串行 I/O 移位寄存器 10 （SIO10）	00H
	串行操作模式寄存器 10 （CSIM10）	00H
	串行时钟选择寄存器 10 （CSIC10）	00H
LCD 控制器/驱动器	LCD 模式设置寄存器 （LCDMD）	00H
	LCD 显示模式寄存器 （LCDM）	00H
	LCD 时钟控制寄存器（LCDC0）	00H
曼彻斯特码发生器	发送缓冲器寄存器 （MC0TX）	FFH
	发送位计数规定计数器（MC0BIT）	07H
	控制寄存器 0 （MC0CTL0）	10H
	控制寄存器 1 （MC0CTL1）	00H
	控制寄存器 2 （MC0CTL2）	1FH
	状态寄存器 （MC0STR）	00H
按键中断	按键返回模式寄存器 （KRM）	00H
复位功能	复位控制标志寄存器 （RESF）	00H [注 2]
低电压检测电路	低电压检测寄存器（LVIM）	00H [注 2]
	低电压检测等级选择寄存器 （LVIS）	00H [注 2]

注　**1.** 在产生复位信号期间或振荡稳定时间等待期间，硬件状态中只有 PC 内容不确定。复位后其它硬件状态保持不变。

2. 这些值的变化取决于复位源。

寄存器＼复位源		$\overline{\text{RESET}}$ 输入	由 POC 进行复位	由 WDT 进行复位	由 LVI 进行复位
RESF	WDTRF 位	清除（0）	清除（0）	置位 （1）	保持
	LVIRF 位			保持	置位 （1）
LVIM		清除（00H）	清除（00H）	清除（00H）	保持
LVIS					

表 23-2. 复位响应后的硬件状态 （4/4）

硬件		响应复位后的状态注 1
遥控器接收器	遥控器接收移位寄存器（RMSR）	00H
	遥控器接收数据寄存器 （RMDR）	00H
	遥控器移位寄存器接受计数器寄存器 （RMSCR）	00H
	遥控器接收 GPHS 比较寄存器（RMGPHS）	00H
	遥控器接收 GPHL 比较寄存器（RMGPHL）	00H
	遥控器接收 DLS 比较寄存器（RMDLS）	00H
	遥控器接收 DLL 比较寄存器（RMDLL）	00H
	遥控器接收 DH0S 比较寄存器（RMDH0S）	00H
	遥控器接收 DH0L 比较寄存器（RMDH0L）	00H
	遥控器接收 DH1S 比较寄存器（RMDH1S）	00H
	遥控器接收 DH1L 比较寄存器（RMDH1L）	00H
	遥控器接收结束宽度选择（RMER）	00H
	遥控器接收中断状态寄存器 （INTS）	00H
	遥控器接收中断状态清除寄存器（INTC）	00H
	遥控器接收控制寄存器（RMCN）	00H
中断	请求标志寄存器 0L，0H，1L，1H （IF0L，IF0H，IF1L，IF1H）	00H
	屏蔽标志寄存器 0L，0H，1L，1H （MK0L，MK0H，MK1L，MK1H）	FFH
	优先级指定标志寄存器 0L，0H，1L，1H （PR0L，PR0H，PR1L，PR1H）	FFH
	外部中断上升沿使能寄存器（EGP）	00H
	外部中断下降沿使能寄存器（EGN）	00H

注　在产生复位信号期间或振荡稳定时间等待期间，硬件状态中只有 PC 内容不确定。复位后其它硬件状态保持不变。

23.1 确认复位源的寄存器

78K0/LE3 中存在许多内部复位产生源。复位控制标志寄存器（RESF）用于存储产生复位请求的复位源。
可以通过 8 位存储器操作指令来读取 RESF。
$\overline{\text{RESET}}$ 输入、由 POC 电路引起的复位输入以及读取 RESF 都会将 RESF 清除为 00H。

图 23-5. 复位控制标志寄存器（RESF）的格式

地址：FFACH　复位后：00H[注]　R

符号	7	6	5	4	3	2	1	0
RESF	0	0	0	WDTRF	0	0	0	LVIRF

WDTRF	看门狗定时器引起的内部复位请求（WDT）
0	未产生内部复位请求，或 RESF 被清除
1	已产生内部复位请求

LVIRF	由低电压检测电路（LVI）产生的内部复位请求
0	未产生内部复位请求，或 RESF 被清除
1	已产生内部复位请求

注　复位后的值的变化取决于复位源。

注意事项　不能通过 1 位存储器操作来指令读取数据。

产生复位请求时 RESF 的状态如表 23-3 所示。

表 23-3. 产生复位请求时 RESF 的状态

<table>
<tr><th>复位源
标志</th><th>$\overline{\text{RESET}}$ 输入</th><th>由 POC 引起的复位</th><th>由 WDT 引起的复位</th><th>由 LVI 引起的复位</th></tr>
<tr><td>WDTRF</td><td rowspan="2">清除（0）</td><td rowspan="2">清除（0）</td><td>置位（1）</td><td>保持</td></tr>
<tr><td>LVIRF</td><td>保持</td><td>置位 （1）</td></tr>
</table>

第二十四章　上电清零电路

24.1 上电清零电路的功能

上电清零电路（POC）具有以下功能。

- 上电时产生内部复位信号。
  在 1.59 V POC 模式下（选项字节： POCMODE = 0），当供电电压（V_{DD}）超过 1.59 V ± 0.15 V 时，释放复位信号。
  在 2.7 V/1.59 V POC 模式下（选项字节： POCMODE = 1），当供电电压（V_{DD}）超过 2.7 V ± 0.2 V 时，释放复位信号。

- 比较供电电压（V_{DD}）和检测电压（V_{POC} = 1.59 V ± 0.15 V），并在 V_{DD} < V_{POC} 时产生内部复位信号。

注意事项　如果在 POC 电路内产生一个内部复位信号，则将复位控制标志寄存器（RESF）被清除为 00H。

备注　78K0/LE3 内置有多种可以产生内部复位信号的硬件功能。RESF 中有一个标志用于指示复位源，当看门狗定时器（WDT）或低电压检测器（LVI）引发内部复位信号时使用。当 WDT 或 LVI 产生内部复位信号时，RESF 不被清除为 00H，标志位被设置为 1。
RESF 的详细内容，参见**第二十三章　复位功能**。

24.2 上电清零电路的配置

上电清零电路的框图如图 24-1 所示。

图 24-1. 上电清零电路的框图

24.3 上电清零电路的操作

（1）　1.59 V POC 模式（选项字节： POCMODE = 0）

- 上电时产生内部复位信号。当供电电压（V_{DD}）超过检测电压（V_{POC} = 1.59 V ± 0.15 V）时，复位状态被释放。
- 比较供电电压（V_{DD}）和检测电压（V_{POC} = 1.59 V ± 0.15 V）。当 $V_{DD} < V_{POC}$ 时，产生内部复位信号。当 $V_{DD} \geq V_{POC}$，释放内部复位信号。

（2）　2.7 V/1.59 V POC 模式（选项字节： POCMODE = 1）

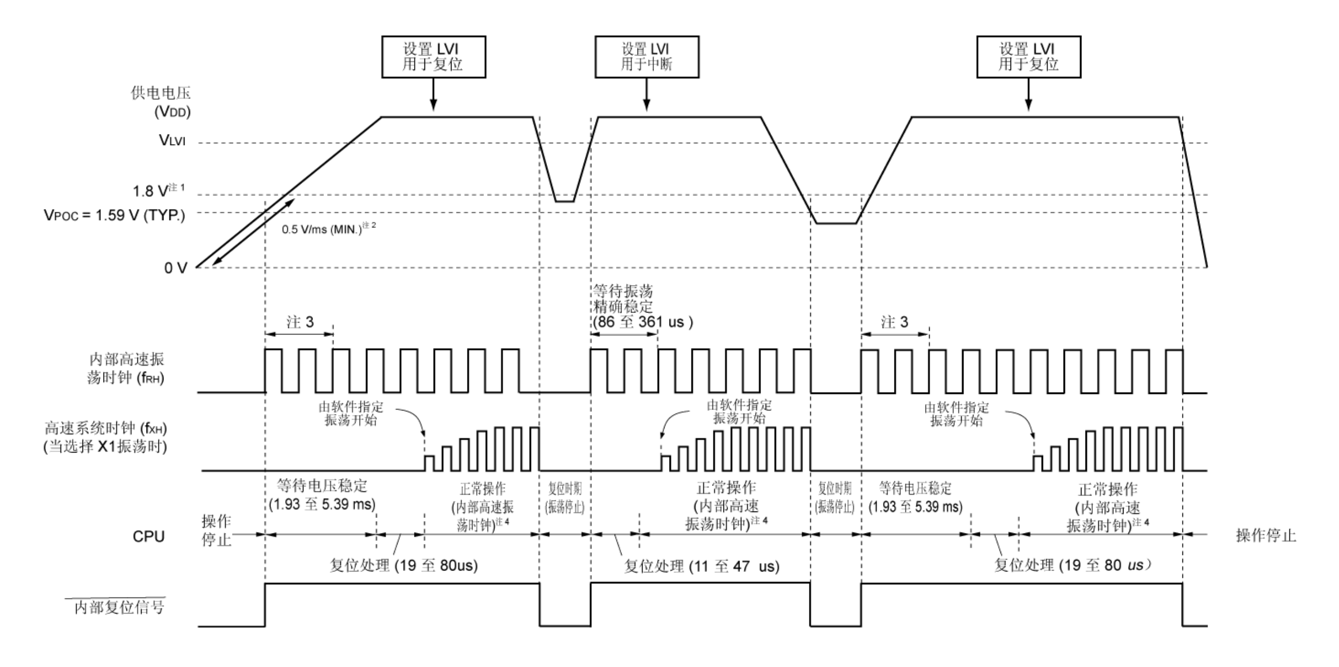
- 上电时产生内部复位信号。当供电电压（V_{DD}）超过检测电压（V_{DDPOC} = 2.7 V ± 0.2 V）时，复位状态被释放。
- 比较供电电压（V_{DD}）和检测电压（V_{POC} = 1.59 V ± 0.15 V）。当 $V_{DD} < V_{POC}$ 时，产生内部复位信号。当 $V_{DD} \geq V_{POC}$ 时，释放内部复位信号。

由上电清零电路和低电压检测器产生内部复位信号的时序如下所示。

<R>

图 24-2. 由上电清零电路和低电压检测器产生内部复位信号的时序（1/2）

（1）　**1.59 V POC 模式（选项字节：POCMODE = 0）**



注　**1.** 操作保证的电压范围是 1.8 V ≤ V_{DD} ≤ 5.5 V。当供电电压下降时，为使状态低于 1.8 V 复位状态，可以使用低电压检测器的复位功能，或通过 $\overline{\text{RESET}}$ 引脚输入低电平。

2. 如果上电后电压上升到 1.8V 的速率小于 0.5 V/ms（最小值），则在上电后且电压抵达 1.8V 之前，输入一个低电平到 $\overline{\text{RESET}}$ 引脚，或者通过选项字节（POCMODE = 1）设置为 2.7 V/1.59 V POC 模式。

3. 内部电压稳定时间包括内部高速振荡时钟的振荡精确稳定时间。

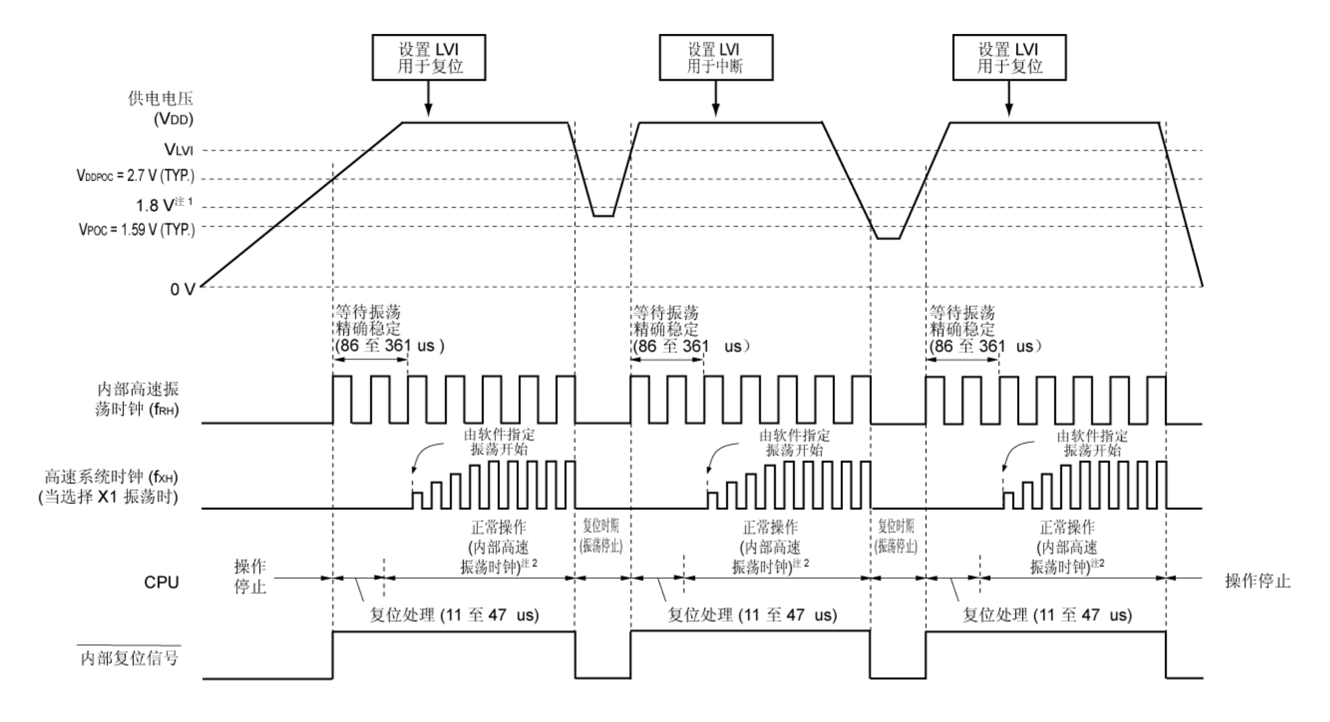
4. CPU 时钟可以选择内部高速振荡时钟和高速系统时钟或副系统时钟。要使用 X1 时钟，可以通过 OSTC 确认是否经过振荡稳定时间。要使用 XT1 时钟，可以通过定时器功能来确认是否经过振荡稳定时间。

注意事项　在释放复位状态后，通过软件设置低电压检测器（参见第二十五章 低电压检测器）。

备注　V_{LVI}：　LVI 检测电压
V_{POC}：　POC 检测电压

图 **24-2.** 由上电清零电路和低电压检测器产生内部复位信号的时序（**2/2**）

（**2**）　**2.7 V/1.59 V POC** 模式（选项字节： **POCMODE = 1**）



注　**1.** 操作保证的电压范围是 1.8 V ≤ V_{DD} ≤ 5.5 V。当供电电压下降时，为使状态低于 1.8 V 复位状态，可以使用低电压检测器的复位功能，或通过 $\overline{\text{RESET}}$ 引脚输入低电平。

2. CPU 时钟可以选择内部高速振荡时钟和高速系统时钟或副系统时钟。要使用 X1 时钟，可以通过 OSTC 确认是否经过振荡稳定时间。要使用 XT1 时钟，可以通过定时器功能来确认是否经过振荡稳定时间。

注意事项 **1.** 在释放复位状态后，通过软件设置低电压检测器（参见第二十五章 低电压检测器）。

<R> **2.** 在电源电压达到 **1.59 V**（典型值）后，需要一个 **1.93** 到 **5.39ms** 的电压振荡稳定时间。如果电源电压在 **1.93ms** 内从 **1.59 V**（典型值）上升到 **2.7 V**（典型值），那么在复位处理之前，自动发生 **0** 到 **5.39ms** 的电源电压振荡稳定时间，且复位处理时间变为 **19** 至 **80** ***μ*s.**。

备注　V_{LVI}：　LVI 检测电压

V_{POC}：　POC 检测电压

24.4 上电清零电路的注意事项

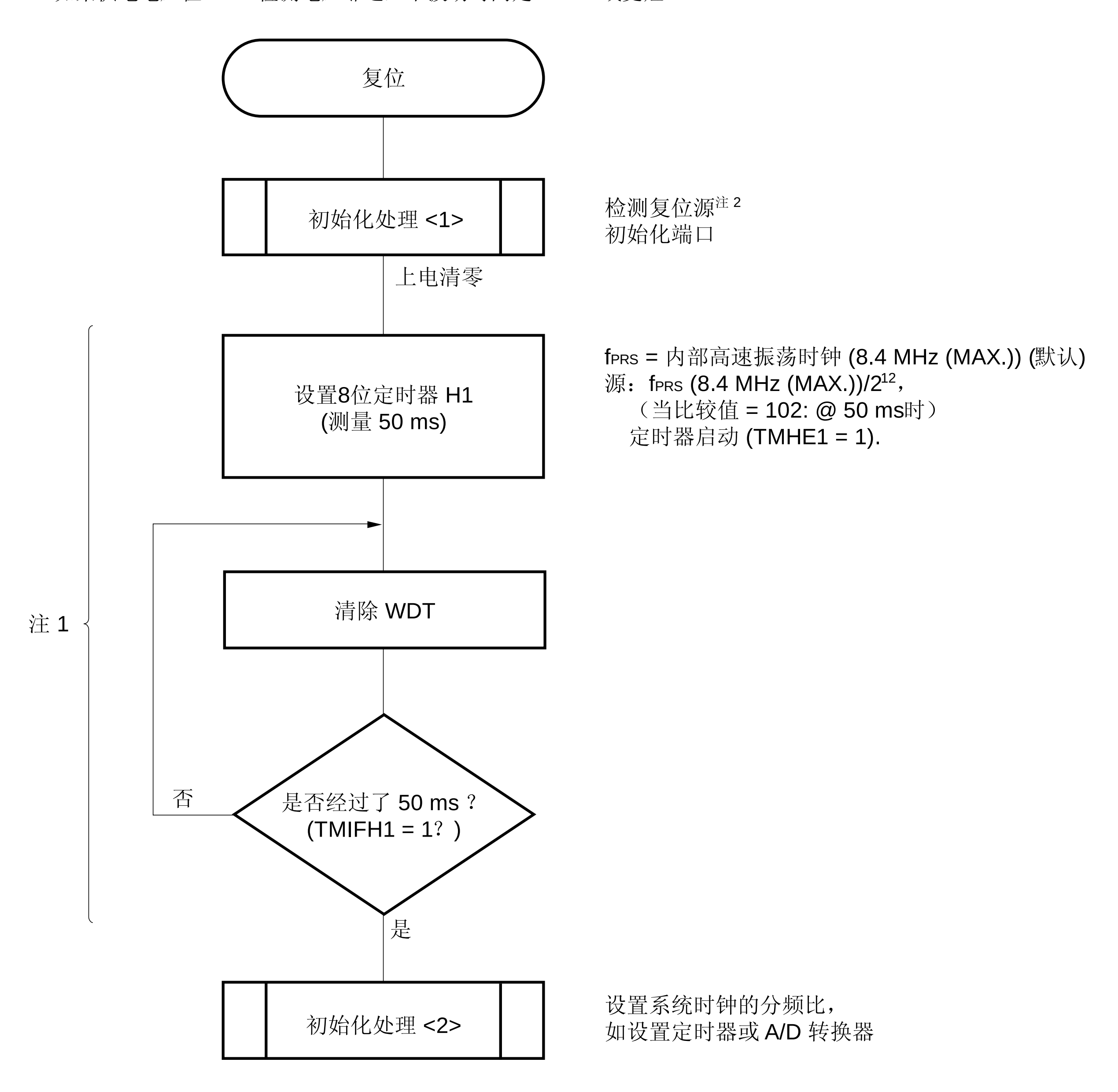
在系统中，如果一定时期内的供电电压（V_{DD}）在 POC 检测电压（V_{POC}）邻近上下波动，则系统可能会反复复位以及从复位状态释放。在这种情况下，可采用下列对策任意设置从复位释放到启动微控制器的操作所需要的时间。

<对策>

在复位信号释放后，通过使用定时器的软件计数器方法等待系统供电电压的波动期，然后初始化端口。

图 24-3. 复位释放后软件处理的示例（1/2）

- 如果供电电压在 POC 检测电压邻近上下波动时间是 50 ms 或更短

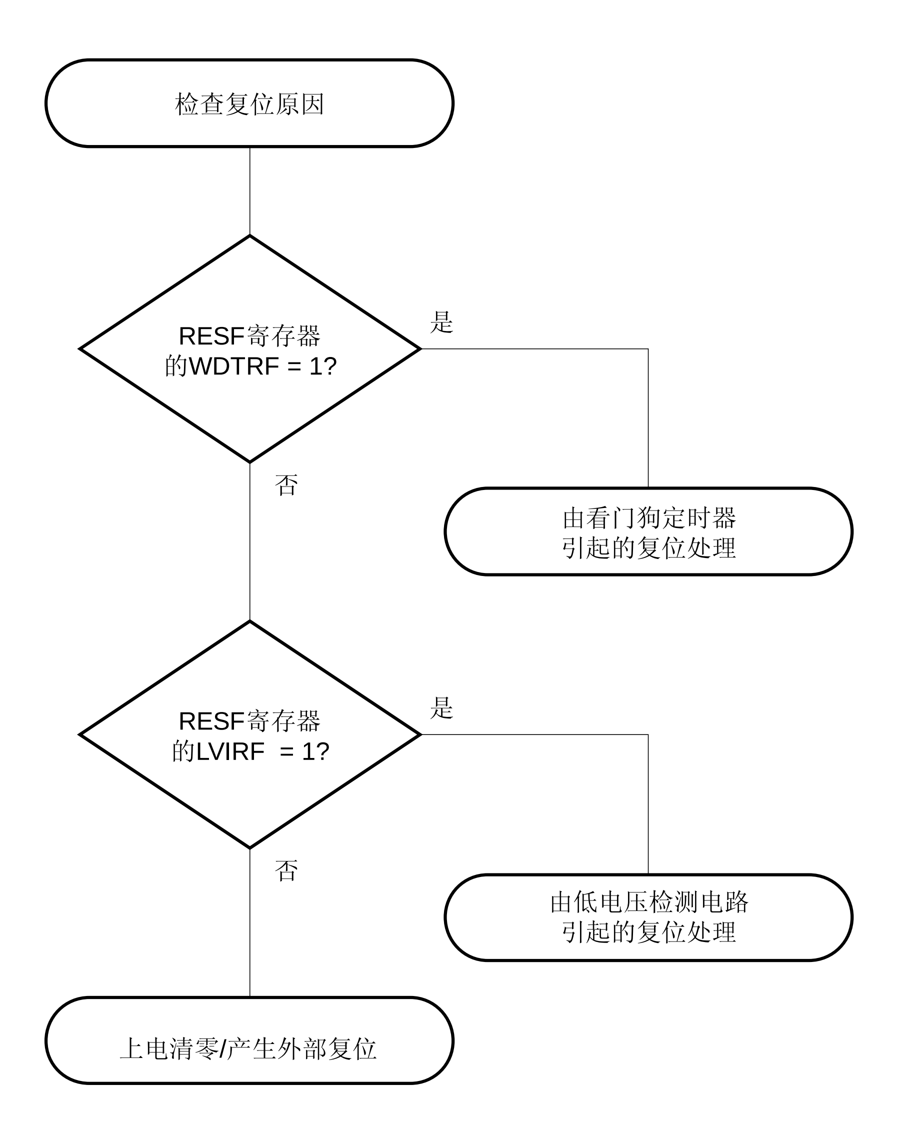


注　**1.** 如果在此期间再次产生复位，则不启动初始化操作<2>。

　　2. 流程图在下页展示。

图 **24-3.** 复位释放后软件处理的示例（**2/2**）

- 检查复位源



第二十五章　低电压检测器

25.1 低电压检测器的功能

低电压检测器（LVI）具有以下功能。

- LVI 电路比较供电电压（V_{DD}）和检测电压（V_{LVI}），或比较外部输入引脚的输入电压（EXLVI）和检测电压（V_{EXLVI} = 1.21 V（典型值）），并产生内部复位或内部中断信号。
- 可由软件选择供电电压（V_{DD}）或外部输入引脚的输入电压（EXLVI）。
- 可由软件选择复位或中断功能。
- 通过软件改变供电电压的检测电平（16 级）。
- 在 STOP 模式下可以操作。

复位和中断信号的产生根据软件的选择，如下所示。

供电电压（V_{DD}）电平检测的选择（LVISEL = 0）		从外部输入引脚（EXLVI）的输入电压电平检测的选择（LVISEL = 1）	
选择复位（LVIMD = 1）.	选择中断（LVIMD = 0）.	选择复位（LVIMD = 1）.	选择中断（LVIMD = 0）.
当 $V_{DD} < V_{LVI}$ 时，产生内部复位信号，并且当 $V_{DD} \geq V_{LVI}$ 时，释放复位信号。	当 V_{DD} 降落低于 V_{LVI}（$V_{DD} < V_{LVI}$）时，或者当 V_{DD} 变为 V_{LVI} 或更高（$V_{DD} \geq V_{LVI}$）时，产生内部中断信号。	当 EXLVI < V_{EXLVI} 时，产生内部复位信号，并且当 EXLVI ≥ V_{EXLVI} 时，释放复位信号。	当 EXLVI 降落低于 V_{EXLVI}（EXLVI < V_{EXLVI}）时，或者当 EXLVI 变为 V_{EXLVI} 或更高（EXLVI ≥ V_{EXLVI}）时，产生内部中断信号。

备注　LVISEL：　低电压检测寄存器（LVIM）的第 2 位
　　　　LVIMD：　LVIM 的第 1 位

当低电压检测器工作时，通过读取低电压检测标志（LVIM 的第 0 位：LVIF），可以检查供电电压或外部输入引脚的输入电压是否高于或低于检测电平。

当低电压检测器用于复位时，如果发生复位，复位控制标志寄存器（RESF）的第 0 位（LVIRF）被设置为 1。RESF 的详细信息，参见**第二十三章 复位功能**。

25.2 低电压检测器的配置

低电压检测器的框图如图 25-1 所示。

图 25-1. 低电压检测器的框图

25.3 控制低电压检测器的寄存器

低电压检测器由以下寄存器控制。

- 低电压检测寄存器（LVIM）
- 低电压检测电平选择寄存器（LVIS）
- 端口模式寄存器 12（PM12）

（1）　低电压检测寄存器 （LVIM）

该寄存器用于设置低电压检测和操作模式。

可以通过 1 位或 8 位存储器操作指令来设置该寄存器。

复位信号的产生会将 LVIM 清除为 00H。

图 25-2. 低电压检测寄存器（LVIM）的格式

地址：FFBEH 复位后：00H [注 1] R/W [注 2]

符号	<7>	6	5	4	3	<2>	<1>	<0>
LVIM	LVION	0	0	0	0	LVISEL	LVIMD	LVIF

LVION [注 3，4]	使能低电压检测操作
0	禁止操作
1	使能操作

LVISEL [注 2]	电压检测的选择
0	检测供电电压（V_{DD}）的电平
1	检测从外部输入引脚（EXLVI）输入的电压的电平

LVIMD [注 3]	低电压检测操作模式（中断/复位）的选择
0	• LVISEL = 0：当 V_{DD} 降落低于 V_{LVI}（$V_{DD} < V_{LVI}$）时，或者当 V_{DD} 变为 V_{LVI} 或更高（$V_{DD} \geq V_{LVI}$）时，产生内部中断信号。 • LVISEL = 1：当 EXLVI 降落低于 V_{EXLVI}（EXLVI < V_{EXLVI}）时，或者当 EXLVI 变为 V_{EXLVI} 或更高（EXLVI ≥ V_{EXLVI}）时，产生内部中断信号。
1	• LVISEL = 0：当 $V_{DD} < V_{LVI}$ 时，产生内部复位信号，并且当 $V_{DD} \geq V_{LVI}$ 时，释放复位信号。 • LVISEL = 1：当 EXLVI < V_{EXLVI} 时，产生内部复位信号，并且当 EXLVI ≥ V_{EXLVI} 时，释放复位信号。

LVIF [注 3]	低电压检测标志
0	• LVISEL = 0：供电电压 （V_{DD}） ≥ 检测电压 （V_{LVI}），或当操作被禁止时 • LVISEL = 1：从外部输入引脚的输入电压（EXLVI） ≥ 检测电压（V_{EXLVI}），或当操作被禁止时
1	• LVISEL = 0：供电电压 （V_{DD}） < 检测电压（V_{LVI}） • LVISEL = 1：从外部输入引脚的输入电压（EXLVI） < 检测电压（V_{EXLVI}）

注　**1.** 在产生 LVI 复位以外的复位时，该位被清除为 00H。

2. 第 0 位只读。

3. 在产生 LVI 复位以外的复位时，LVION、LVIMD 和 LVISEL 被清除为 0。在产生 LVI 复位时，这些位不被清除为 0。

4. 当 LVION 被设置为 1 时，LVI 电路中的比较器开始操作。使用软件等待操作稳定时间（10 μs（最大值））从 LVION 被设置为 1 到操作稳定。在操作已经稳定后，从进入一个低于 LVI 检测电压的状态直到 LVIF 被置位（1），需要 200 μs（最小值）。

注意事项　1. 按照下列任一过程，可以停止 LVI。

- **当使用 8 位存储器操作指令时：写入 00H 到 LVIM。**
- **当使用 1 位存储器操作指令时：将 LVION 清除为 0。**

2. 从外部输入引脚（EXLVI）输入的电压必须满足 EXLVI < V_{DD}。

3. 当 LVI 用于中断时，如果在 LVI 检测电压状态下，LVION 被清 0 ，会产生 INTLVI 信号并将 LVIIF 置 1。

（2）　低电压检测电平选择寄存器（LVIS）

该寄存器用于选择低电压检测电平。

可以通过 1 位或 8 位存储器操作指令来设置该寄存器。

产生 LVI 复位之外的复位信号会将 LVIS 清除为 00H。

图 25-3. 低电压检测电平选择寄存器（LVIS）的格式

地址：FFBFH　复位后：00H 注　R/W

符号	7	6	5	4	3	2	1	0
LVIS	0	0	0	0	LVIS3	LVIS2	LVIS1	LVIS0

LVIS3	LVIS2	LVIS1	LVIS0	检测电平
0	0	0	0	V_{LVI0}（4.24 V ± 0.1 V）
0	0	0	1	V_{LVI1}（4.09 V ± 0.1 V）
0	0	1	0	V_{LVI2}（3.93 V ± 0.1 V）
0	0	1	1	V_{LVI3}（3.78 V ± 0.1 V）
0	1	0	0	V_{LVI4}（3.62 V ± 0.1 V）
0	1	0	1	V_{LVI5}（3.47 V ± 0.1 V）
0	1	1	0	V_{LVI6}（3.32 V ± 0.1 V）
0	1	1	1	V_{LVI7}（3.16 V ± 0.1 V）
1	0	0	0	V_{LVI8}（3.01 V ± 0.1 V）
1	0	0	1	V_{LVI9}（2.85 V ± 0.1 V）
1	0	1	0	V_{LVI10}（2.70 V ± 0.1 V）
1	0	1	1	V_{LVI11}（2.55 V ± 0.1 V）
1	1	0	0	V_{LVI12}（2.39 V ± 0.1 V）
1	1	0	1	V_{LVI13}（2.24 V ± 0.1 V）
1	1	1	0	V_{LVI14}（2.08 V ± 0.1 V）
1	1	1	1	V_{LVI15}（1.93 V ± 0.1 V）

注　由 LVI 引起的复位时，LVIS 的值并不被重置，而是保留其值。其它原因引起的复位会将其清除为 00H。

注意事项 **1.** **请确保将第 4 位至第 7 位清除为“0”。**

2. LVI 操作期间，不要改变 LVIS 的值。

3. 当检测外部输入引脚（EXLVI）输入的电压时，检测电压（V_{EXLVI} = 1.21 V（典型值））被固定。因此，无需设置 LVIS。

（3）　端口模式寄存器 12（PM12）

当使用 P120/EXLVI/INTP0 引脚作为外部低电压检测的电位输入时，设置 PM120 为 1。此时，P120 的输出锁存器的值可能是 0 或 1。

可以通过 1 位或 8 位存储器操作指令来设置 PM12。

复位信号的产生会将 PM12 设置为 FFH。

图 25-4. 端口模式寄存器 12（PM12）的格式

地址：FF2CH 复位后：FFH R/W

符号	7	6	5	4	3	2	1	0
PM12	1	1	1	1	1	1	1	PM120

PM120	P120 引脚 I/O 模式选择
0	输出模式（输出缓冲器打开）
1	输入模式（输出缓冲器关闭）

25.4 低电压检测器的操作

低电压检测器可以用于以下两种模式。

（1）　用作复位（LVIMD = 1）

- 如果 LVISEL = 0，比较供电电压（V_{DD}）与检测电压（V_{LVI}），当 $V_{DD} < V_{LVI}$ 时，产生内部复位信号，而当 $V_{DD} \geq V_{LVI}$ 时，释放内部复位。
- 如果 LVISEL = 1，比较外部输入引脚的输入电压（EXLVI）和检测电压（V_{EXLVI} = 1.21 V（典型值）），在 EXLVI < V_{EXLVI} 时，产生内部复位信号，而在 EXLVI ≥ V_{EXLVI} 时，释放内部复位。

（2）　用于中断（LVIMD = 0）

- 如果 LVISEL = 0，比较供电电压（V_{DD}）与检测电压（V_{LVI}），当 $V_{DD} < V_{LVI}$ 时产生中断信号（INTLVI）。
- 如果 LVISEL = 1，比较外部输入引脚的输入电压（EXLVI）和检测电压（V_{EXLVI} = 1.21 V（典型值）），当 EXLVI < V_{EXLVI} 时产生中断信号（INTLVI）。

当低电压检测器工作时，通过读取低电压检测标志（LVIM 的第 0 位：LVIF），可以检查供电电压或外部输入引脚的输入电压是否高于或低于检测电平。

备注　LVIMD：　低电压检测寄存器（LVIM）的第 1 位
　　　　LVISEL：　LVIM 的第 2 位

25.4.1 用于复位

（1）　当检测供电电压（V_{DD}）的电平时

- 启动操作时

  <1>　屏蔽 LVI 中断 （LVIMK = 1）。

  <2>　将低电压检测寄存器（LVIM）的第 2 位（LVISEL）清除为 0（检测供电电压（V_{DD}）的电平）（默认值）。

  <3>　使用低电压检测电平选择寄存器（LVIS）的第 3 位至第 0 位（LVIS3 至 LVIS0）来设置检测电压。

  <4>　设置 LVIM 的第 7 位（LVION）为 1（使能 LVI 的操作）。

  <5>　使用软件来等待操作稳定时间（10 μs（最大值））。

  <6>　等待，直到通过 LVIM 的第 0 位（LVIF）检测到供电电压（V_{DD}） ≥ 检测电压（V_{LVI}）。

  <7>　设置 LVIM 的第 1 位（LVIMD）为 1（当检测到该电平时，产生复位）。

  图 25-5 展示了由低电压检测器产生的内部复位信号的时序。该时序图中的标号与上述的<1>至<7>对应。

  注意事项　1.　必须执行过程<1>。当 LVIMK = 0 时，在过程<4>之后可能会立即产生中断。

  2.　当 LVIMD 被设置为 1 时，如果供电电压（V_{DD}） ≥ 检测电压（V_{LVI}），则不会产生内部复位信号。

- 当停止操作时

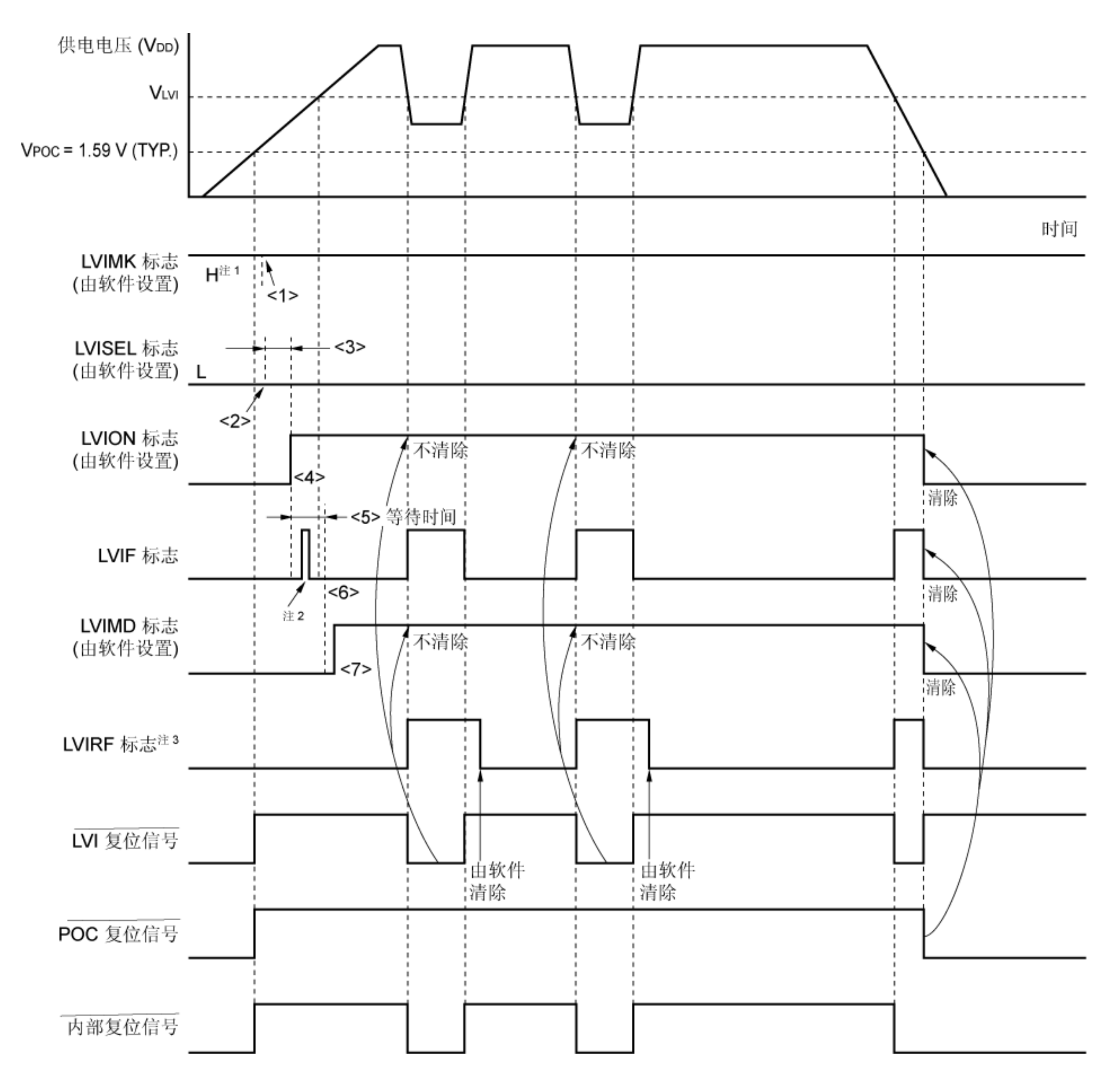
  必须执行以下过程之一。

  - 当使用 8 位存储器操作指令时：
    向 LVIM 写入 00H。

  - 当使用 1 位存储器操作指令时：
    将 LVIMD 清除为 0，然后将 LVION 清除为 0。

图 25-5. 低电压检测器内部复位信号的产生时序
（检测供电电压（V_{DD}）的电平）（1/2）

（1）　1.59 V POC 模式（选项字节：POCMODE = 0）

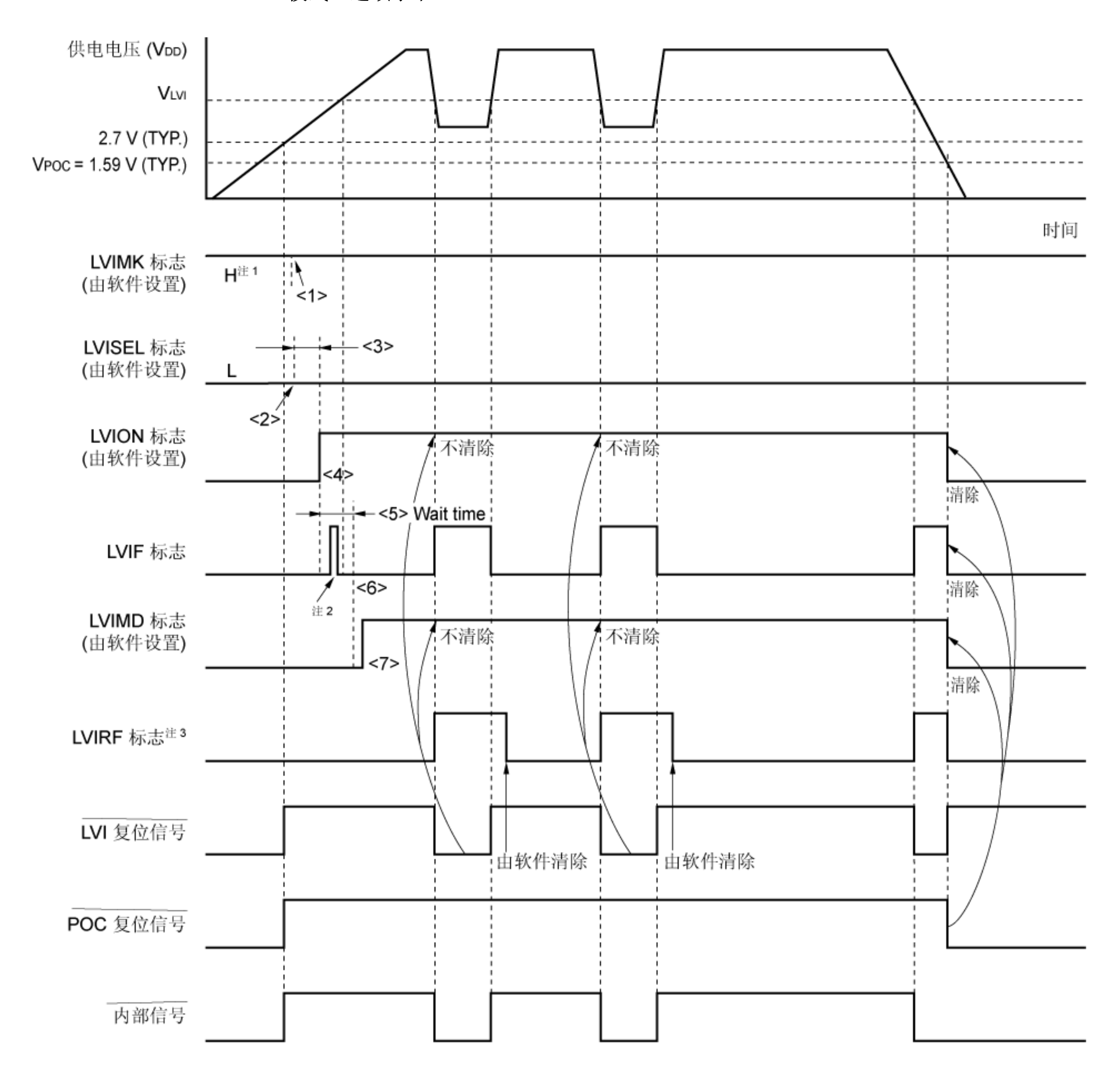


注　**1.**　复位信号的产生会将 LVIMK 标志设置为“1”。
　　2.　LVIF 标志可能被置位（1）。
　　3.　LVIRF 是复位控制标志寄存器（RESF）的第 0 位。RESF 的详细信息，参见第二十三章　复位功能。

备注　图 25-5 中的<1>至<7>对应于 **25.4.1（1）　当检测供电电压（V_{DD}）的电平时**“当启动操作时”中<1>至<7>的描述。

图 **25-5.** 低电压检测器内部复位信号的产生时序
（检测供电电压（V_{DD}）的电平）（**2/2**）

（**2**）　**2.7 V/1.59 V POC** 模式（选项字节：**POCMODE = 1**）



注　**1.** 复位信号的产生会将 LVIMK 标志设置为“1”。
　　2. LVIF 标志可能被置位（1）。
　　3. LVIRF 是复位控制标志寄存器（RESF）的第 0 位。RESF 的详细信息，参见第二十三章 复位功能。

备注　图 25-5 中的<1>至<7>对应于 **25.4.1（1） 当检测供电电压（V_{DD}）的电平时**“当启动操作时”中<1>至<7>的描述。

（2）　当检测外部输入引脚输入电压（EXLVI）的电平时

- 当启动操作时

<1>　屏蔽 LVI 中断（LVIMK = 1）。

<2>　设置低电压检测寄存器（LVIM）的第 2 位（LVISEL）为 1（检测外部输入引脚输入电压（EXLVI）的电平）。

<3>　设置 LVIM 的第 7 位（LVION）为 1（使能 LVI 的操作）。

<4>　使用软件来等待操作稳定时间（10 μs（最大值））。

<5>　等待，直到通过 LVIM 的第 0 位（LVIF）检测到外部输入引脚的输入电压（EXLVI） ≥ 检测电压（V_{EXLVI} = 1.21 V（典型值））

<6>　设置 LVIM 的第 1 位（LVIMD）为 1（当检测到该电平时，产生复位）。

图 25-5 展示了由低电压检测器产生的内部复位信号的时序。该时序图中的标号与上述的<1>至<6>对应。

注意事项　**1.　必须执行过程<1>。当 LVIMK = 0 时，在过程<3>之后可能会立即产生中断。**

2.　当 LVIMD 被设置为 1 时，如果外部输入引脚的输入电压（EXLVI） ≥ 检测电压（V_{EXLVI} = 1.21 V（典型值）），则不会产生内部复位信号。

3.　外部输入引脚（EXLVI）的输入电压必须满足 EXLVI < V_{DD}。

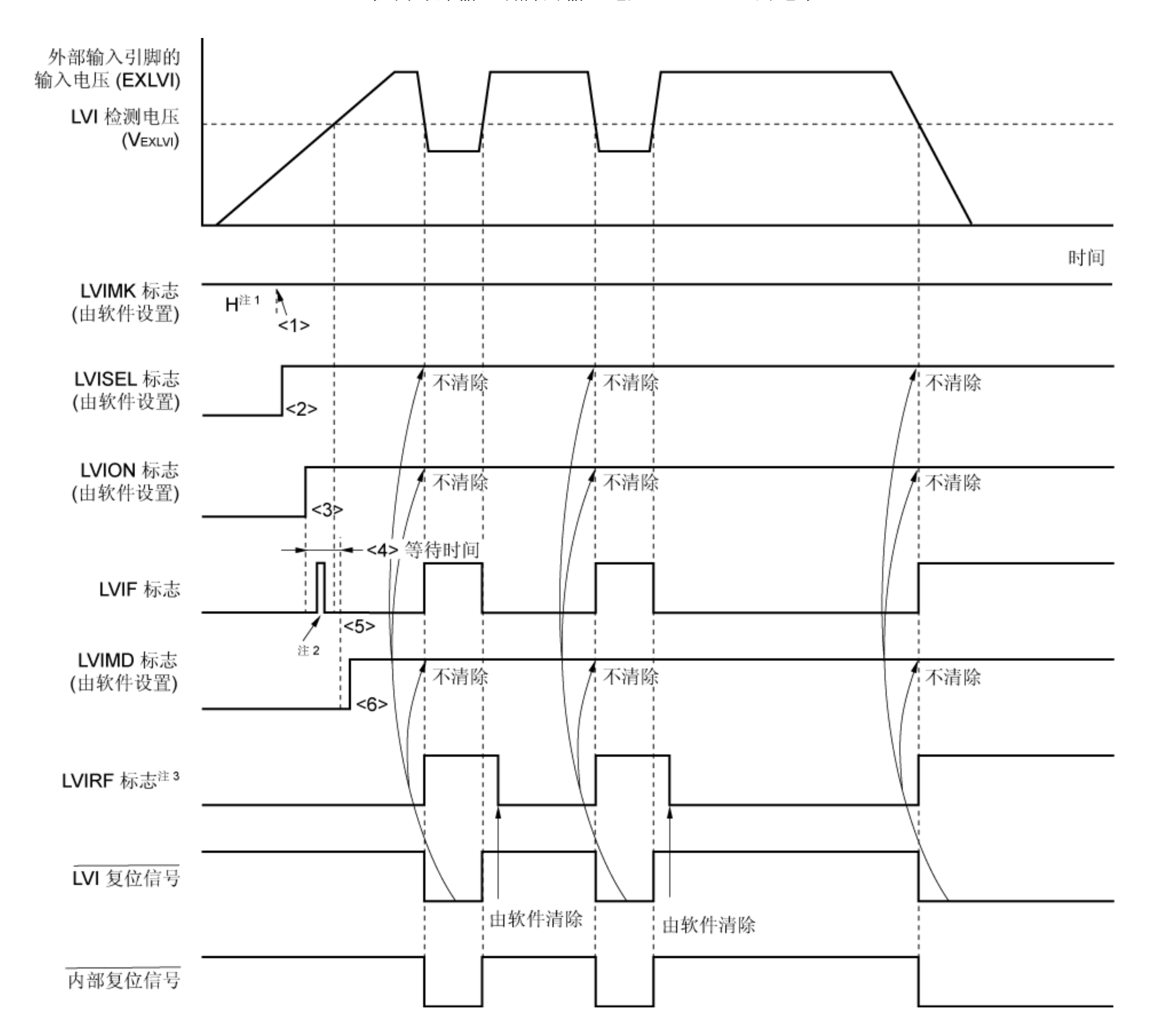
- 当停止操作时

必须执行以下过程之一。

- 当使用 8 位存储器操作指令时：
  向 LVIM 写入 00H。

- 当使用 1 位存储器操作指令时：
  将 LVIMD 清除为 0，然后将 LVION 清除为 0。

图 **25-6.** 低电压检测器内部复位信号的产生时序
（检测外部输入引脚的输入电压（**EXLVI**）的电平）



注　**1.** 复位信号的产生会将 LVIMK 标志设置为“1”。

2. LVIF 标志可能被置位（1）。

3. LVIRF 是复位控制标志寄存器（RESF）的第 0 位。RESF 的详细信息，参见**第二十三章　复位功能**。

备注　图 25-6 中的<1>至<6>对应于 **25.4.1（2）　当检测外部输入引脚的输入电压（EXLVI）电平时**“当启动操作时”中<1>至<6>的描述。

25.4.2 用于中断

（1）　当检测供电电压（V_{DD}）电平时

- 当启动操作时

  <1>　屏蔽 LVI 中断 （LVIMK = 1）
  <2>　将低电压检测寄存器（LVIM）的第 2 位（LVISEL）清除为 0（检测供电电压（V_{DD}）电平）（默认值）。
  <3>　使用低电压检测电平选择寄存器（LVIS）的第 3 位至第 0 位（LVIS3 至 LVIS0）来设置检测电压。
  <4>　设置 LVIM 的第 7 位（LVION）为 1（使能 LVI 的操作）。
  <5>　使用软件来等待操作稳定时间（10μs（最大值））。
  <6>　通过 LVIM 的第 0 位（LVIF），当检测 V_{DD} 的下降沿时，确认“供电电压（V_{DD}） ≥ 检测电压（V_{LVI}）”，或者检测 V_{DD} 的上升沿时，确认“供电电压（V_{DD}） < 检测电压（V_{LVI}）”。
  <7>　将 LVI 的中断请求标志（LVIIF）清除为 0。
  <8>　释放 LVI 的中断屏蔽标志（LVIMK）。
  <9>　将 LVIM 的第 1 位（LVIMD）清除为 0（当检测到该电平时，产生中断信号）（默认值）。
  <10> 执行 EI 指令（当使用向量中断时）。

  图 25-7 展示了由低电压检测器产生中断信号的时序。该时序图中的标号与上述的<1>至<9>对应。

- 当停止操作时

  必须执行以下过程之一。

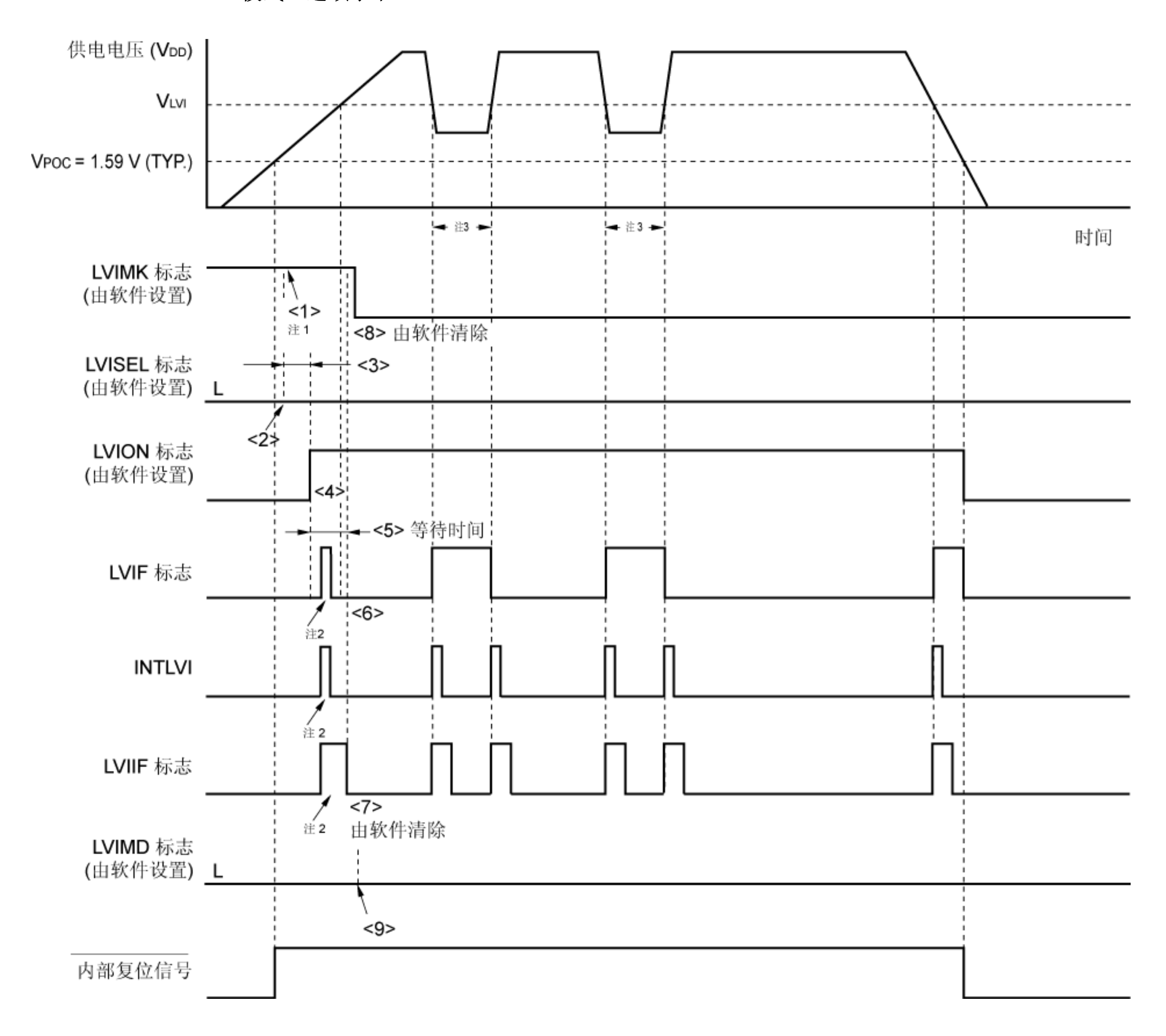
  - 当使用 8 位存储器操作指令时：
    向 LVIM 写入 00H。

  - 当使用 1 位存储器操作指令时：
    将 LVION 清除为 0。

图 **25-7.** 低电压检测器中断信号的产生时序

（检测供电电压（**V**DD）的电平）（**1/2**）

（**1**） **1.59 V POC** 模式（选项字节： **POCMODE = 0**）



注 **1.** 复位信号的产生会将 LVIMK 标志设置为“1”。

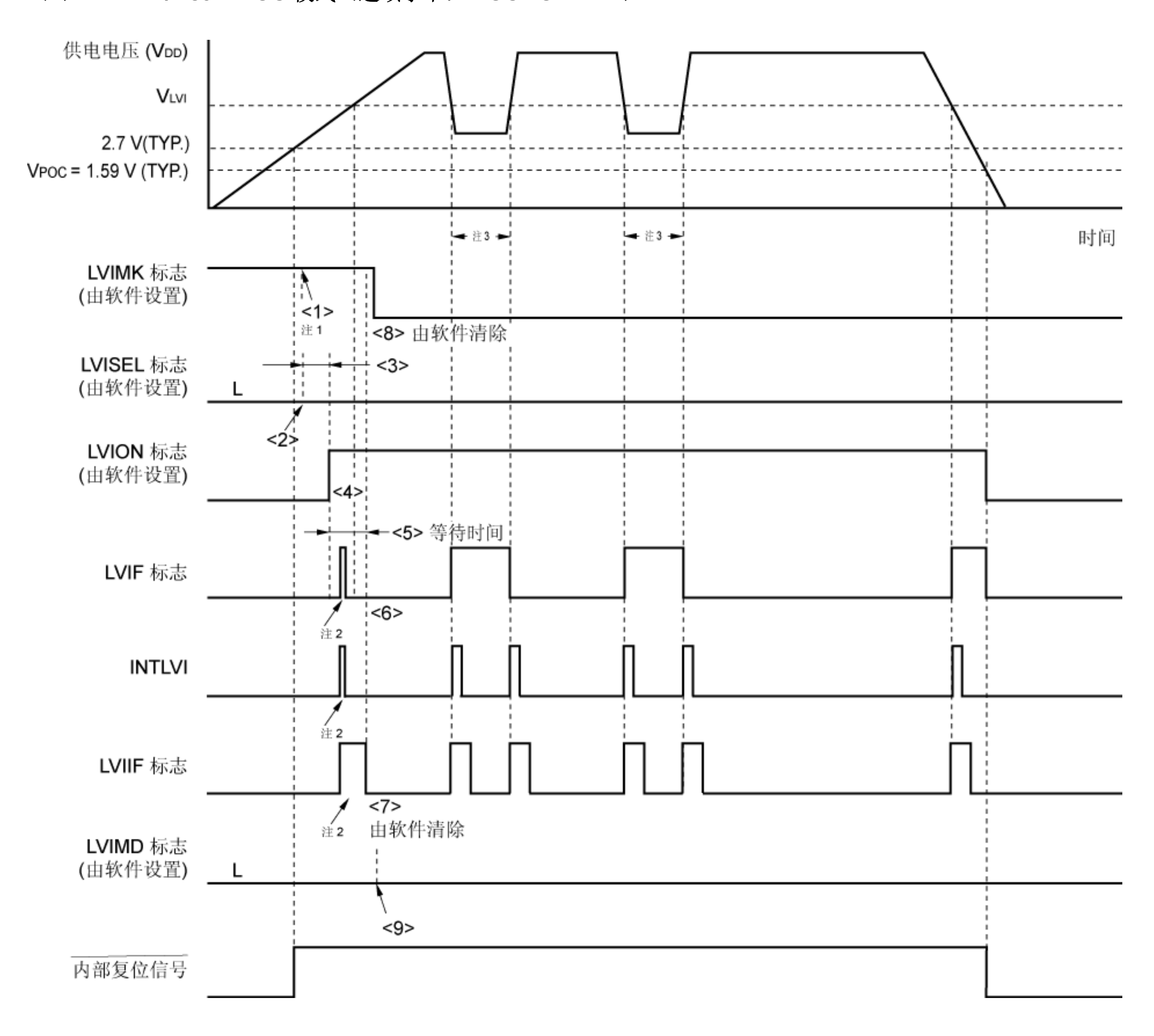
2. 产生中断请求信号（INTLVI），且 LVIF 标志可能被置位（1）。

3. 如果在低于 LVI 检测电压的状态下将 LVION 清除（0），那么产生 INTLVI 信号，且 LVIIF 变成 1。

备注 图 25-7 中的<1>至<9>对应于 **25.4.2**（**1**） **当检测供电电压（V**DD**）的电平时**“当启动操作时”中<1>至<9>的描述。

图 **25-7.** 低电压检测器中断信号的产生时序
（检测供电电压（V_{DD}）的电平）（**2/2**）

（**2**）　**2.7 V/1.59 V POC** 模式（选项字节： **POCMODE = 1**）



注　**1.** 复位信号的产生会将 LVIMK 标志设置为“1”。

2. 产生中断请求信号（INTLVI），且 LVIF 标志可能被置位（1）。

3. 如果在低于 LVI 检测电压的状态下将 LVION 清除（0），那么产生 INTLVI 信号，且 LVIIF 变成 1。

备注　图 25-7 中的<1>至<9>对应于 **25.4.2**（**1**） **当检测供电电压**（V_{DD}）**的电平时**“当启动操作时”中<1>至<9>的描述。

（2）　当检测外部输入引脚的输入电压（EXLVI）电平时

- 当启动操作时

<1>　屏蔽 LVI 中断（LVIMK = 1）

<2>　设置低电压检测寄存器（LVIM）的第 2 位（LVISEL）为 1（检测外部输入引脚的输入电压（EXLVI）电平）。

<3>　设置 LVIM 的第 7 位（LVION）为 1（使能 LVI 的操作）。

<4>　使用软件来等待操作稳定时间（10μs（最大值））。

<5>　通过 LVIM 的第 0 位（LVIF），当检测 V_{DD} 的下降沿时，确认“外部输入引脚的输入电压（EXLVI）≥ 检测电压（V_{EXLVI} = 1.21 V（典型值））”，或者检测 V_{DD} 的上升沿时，确认“外部输入引脚的输入电压（EXLVI）< 检测电压（V_{EXLVI} = 1.21 V（典型值））”。

<6>　将 LVI 的中断请求标志（LVIIF）清除为 0。

<7>　释放 LVI 的中断屏蔽标志（LVIMK）。

<8>　将 LVIM 的第 1 位（LVIMD）清除为 0（当检测到该电平时，产生中断信号）（默认值）。

<9>　执行 EI 指令（当使用向量中断时）。

图 25-8 展示了由低电压检测器产生中断信号的时序。该时序图中的标号与上述的<1>至<8>对应。

注意事项　外部输入引脚（EXLVI）的输入电压必须满足 EXLVI < V_{DD}。

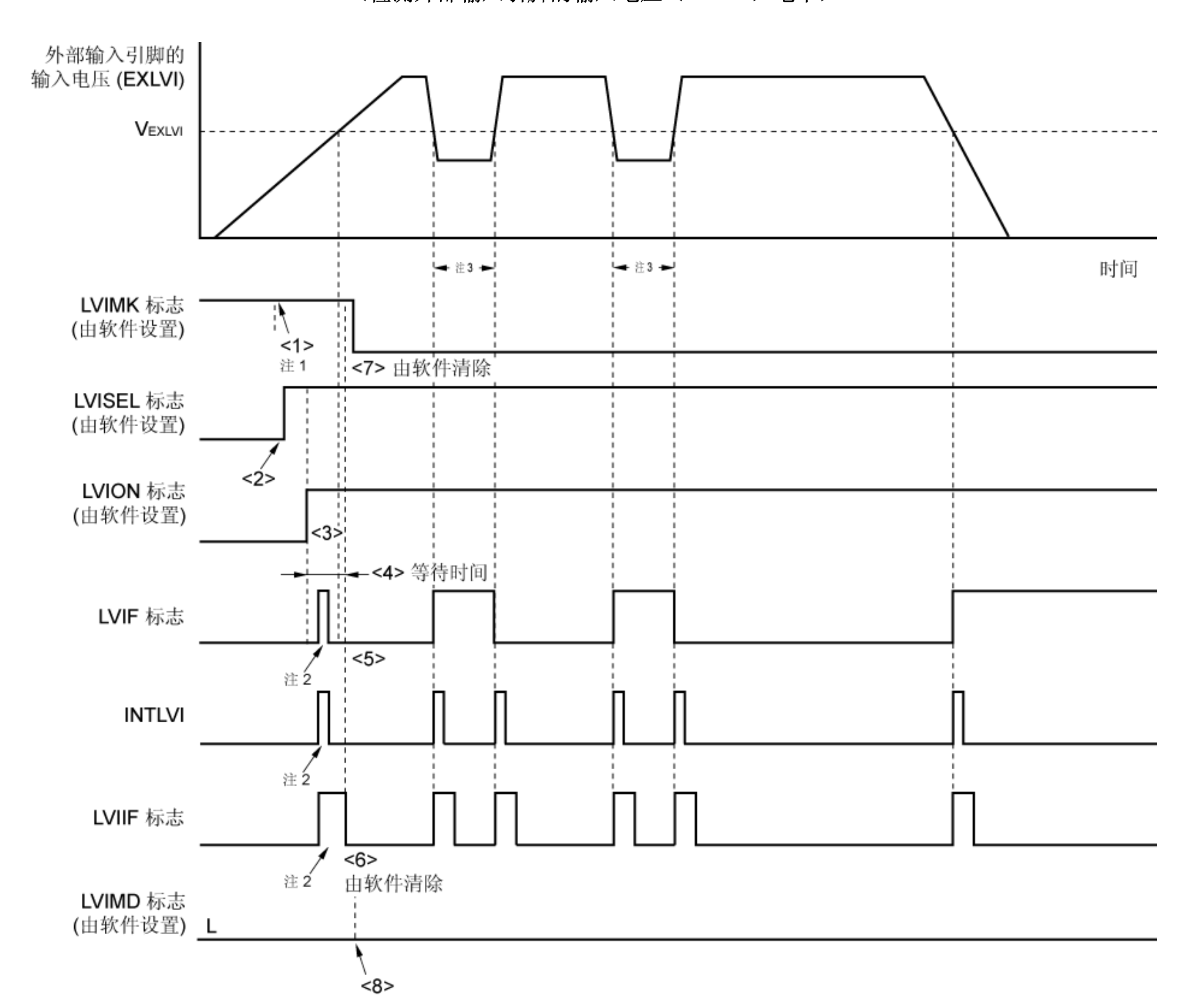
- 当停止操作时

必须执行以下过程之一。

  - 当使用 8 位存储器操作指令时：
    向 LVIM 写入 00H。

  - 当使用 1 位存储器操作指令时：
    将 LVION 清除为 0。

图 **25-8.** 低电压检测器中断信号的产生时序
（检测外部输入引脚的输入电压（**EXLVI**）电平）



注　**1.** 复位信号的产生会将 LVIMK 标志设置为“1”。

2. 产生中断请求信号（INTLVI），且 LVIF 和 LVIIF 标志可能被置位（1）。

3. 如果在低于 LVI 检测电压的状态下将 LVION 清除（0），那么产生 INTLVI 信号，且 LVIIF 变成 1。

备注　图 25-8 中的<1>至<8>对应于 **25.4.2（2） 当检测外部输入引脚的输入电压（EXLVI）电平时**“当启动操作时”中<1>至<8>的描述。

25.5 低电压检测器的注意事项

在系统中，如果一定时期内的供电电压（V_{DD}）在 LVI 检测电压（V_{LVI}）邻近上下波动，这时根据低电压检测器的使用情况进行如下操作。

（1） 用于复位

系统可能会反复进行复位和从复位状态释放。

在这种情况下，可采用下面的对策（1）任意设置从复位释放到启动微控制器的操作所需要的时间。

（2） 用于中断

可能会频繁产生中断请求。可采用对策（2）中的（b）。

<方法>

（1） 用于复位时

在释放复位信号后，通过使用定时器的软件计数器方法等待系统供电电压的波动期，然后初始化端口。（参见**图 25-9**）。

（2） 用于中断时

（a） 在 LVI 中断服务程序中，通过低电压检测寄存器（LVIM）的第 0 位（LVIF），当检测 V_{DD} 的下降沿时，确认“供电电压（V_{DD}） ≥ 检测电压（V_{LVI}）”，或者检测 V_{DD} 的上升沿时，确认“供电电压（V_{DD}） < 检测电压（V_{LVI}）”。中断请求标志寄存器 0L（IF0L）的第 0 位（LVIIF）被清除为 0。

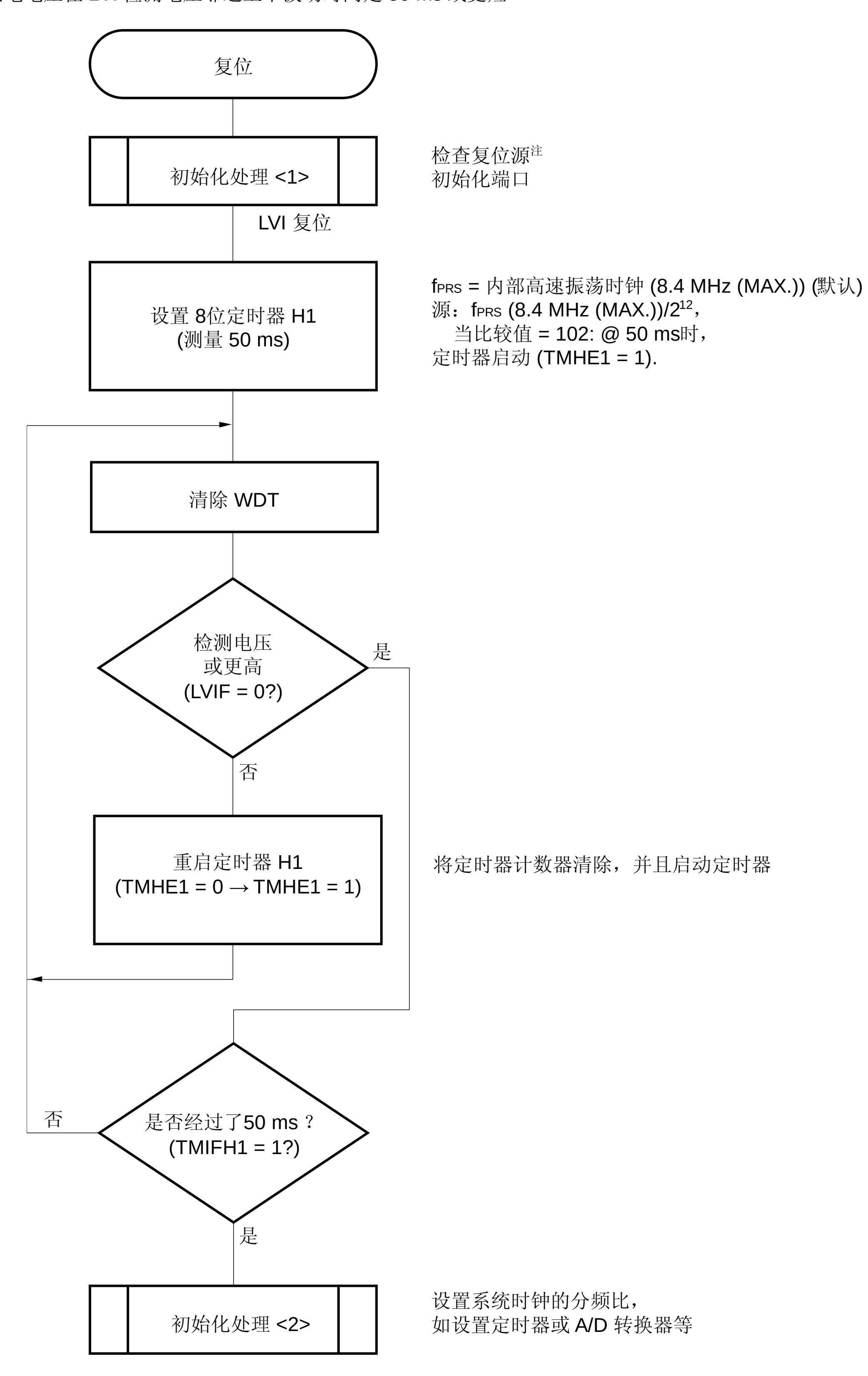
（b） 在系统中，如果一定时期内的供电电压（V_{DD}）在 LVI 检测电压（V_{LVI}）邻近上下波动，等待供电电压波动时期，使用 LVIF 标志，当检测 V_{DD} 的下降沿时，确认“供电电压（V_{DD}） ≥ 检测电压（V_{LVI}）”，或者检测 V_{DD} 的上升沿时，确认“供电电压（V_{DD}） < 检测电压（V_{LVI}）”。并将 LVIIF 标志清除为 0。

备注 如果设置低电压检测寄存器（LVIM）的第 2 位（LVISEL）被设置为“1”，则对上述方法作如下修改。

- 供电电压（V_{DD}） → 外部输入引脚（EXLVI）的输入电压
- 检测电压（V_{LVI}） → 检测电压（V_{EXLVI} = 1.21 V）

图 25-9. 复位释放后软件处理的示例（1/2）

• 如果供电电压在 LVI 检测电压邻近上下波动时间是 50 ms 或更短

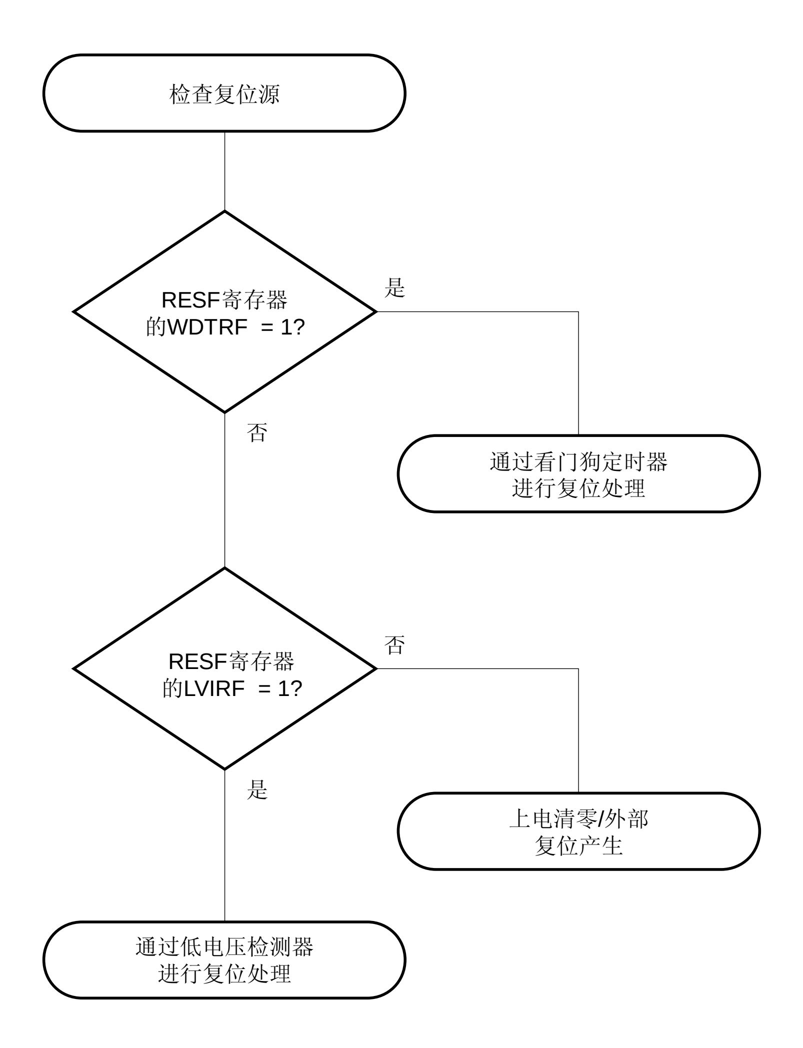


注　　流程图在下页。

<R>

图 25-9. 复位释放后软件处理的示例（2/2）

- 检查复位源



第二十六章　选项字节

26.1 选项字节的功能

78K0/LE3 中 Flash 存储器的 0080H 至 0084H 是选项字节区域。当打开电源或从复位状态重启设备时，设备自动参考选项字节，并设置指定的功能。使用该产品时，请确保使用选项字节设置以下几项功能。

在自编程期间使用引导交换功能时，0080H 至 0084H 被切换到 1080H 至 1084H。因此，事先将 0080H 至 0084H 和 1080H 至 1084H 设置为相同的值。

注意事项　请确保将 0082H 和 0083H（当使用引导交换功能时，是 0082H/1082H 和 0083H/1083H）设置为 00H。

（1）　0080H/1080H

- ○ 内部低速振荡器的操作
  - 可由软件停止
  - 不能停止
- ○ 看门狗定时器间隔时间的设置
- ○ 看门狗定时器计数器的操作
  - 使能计数器的操作
  - 禁止计数器的操作
- ○ 看门狗定时器窗口打开时期的设置

注意事项　将 0080H 和 1080H 设置成为相同的值，因为在引导交换期间 0080H 和 1080H 将被交换。

（2）　0081H/1081H

- ○ 选择 POC 模式
  - 2.7 V/1.59 V POC 模式操作期间（POCMODE = 1）
    上电时该设备处于复位状态，直至供电电压达到 2.7 V（典型值）。当电压超过 2.7 V（典型值）时，设备从复位状态释放。此后，在 2.7 V 不检测 POC，而是在 1.59 V（典型值）检测 POC。
    如果上电后供电电压以低于 0.5 V/ms（最小值）的速率上升到 1.8 V，推荐使用 2.7 V/1.59 V POC 模式。
  - 1.59 V POC 模式操作期间（POCMODE = 0）
    上电时该设备处于复位状态，直至供电电压达到 1.59 V（典型值）。当电压超过 1.59 V（典型值）时，设备从复位状态释放。此后，采用与上电时相同的方式在 1.59 V（典型值）检测 POC。

注意事项　POCMODE 只能通过专用的 Flash 存储器编程器写入。在自编程或是自编程的引导交换操作中不能被设置（此时，设置为 1.59 V POC 模式（默认））。但是，因为在引导交换过程中 1081H 的值被拷贝到 0081H，所以在使用引导交换功能时，推荐将 0081H 和 1081H 设定为相同的值。

（3） 0084H/1084H

○ 片上调试操作控制

- 禁止片上调试操作
- 在片上调试安全 ID 的验证失效的情况下，使能片上调试操作，并且擦除 Flash 存储器的数据。
- 在片上调试安全 ID 的验证失效的情况下，使能片上调试操作，但不擦除 Flash 存储器的数据。

注意事项 要片上调试功能，需要将 0084H 设置为 02H 或 03H。因为在引导交换过程中 0084H 和 1084H 被交换，因此也要对 1084H 设置和 0084H 相同的值。

26.2 选项字节的格式

选项字节格式如下所示。

图 26-1. 选项字节的格式（1/2）

地址： 0080H/1080H 注

7	6	5	4	3	2	1	0
0	WINDOW1	WINDOW0	WDTON	WDCS2	WDCS1	WDCS0	LSROSC

WINDOW1	WINDOW0	看门狗定时器窗口打开时期
0	0	25%
0	1	50%
1	0	75%
1	1	100%

WDTON	看门狗定时器计数器的操作控制/非法访问检测
0	禁止计数器操作（复位后停止计数），禁止非法访问检测操作
1	使能计数器操作（复位后开始计数），使能非法访问检测操作

WDCS2	WDCS1	WDCS0	看门狗定时器溢出时间
0	0	0	$2^{10}/f_{RL}$（3.88 ms）
0	0	1	$2^{11}/f_{RL}$（7.76 ms）
0	1	0	$2^{12}/f_{RL}$（15.52 ms）
0	1	1	$2^{13}/f_{RL}$（31.03 ms）
1	0	0	$2^{14}/f_{RL}$（62.06 ms）
1	0	1	$2^{15}/f_{RL}$（124.12 ms）
1	1	0	$2^{16}/f_{RL}$（248.24 ms）
1	1	1	$2^{17}/f_{RL}$（496.48 ms）

LSROSC	内部低速振荡器操作
0	可由软件停止（当 1 被写入 RCM 寄存器的第 0 位（LSRSTOP）时，停止操作）
1	不能停止（即使将 1 写入 LSRSTOP 位，也不停止操作）

注 因为在引导交换时 0080H 和 1080H 要相互交换，因此要将 0080H 和 1080H 设置为相同的值。

注意事项
1. **禁止设置 WDCS2 = WDCS1 = WDCS0 = 0 且 WINDOW1 = WINDOW0 = 0。**
2. **Flash 存储器自编程和 EEPROM 模拟期间，看门狗定时器继续工作。在处理期间，中断响应时间被延迟。设置溢出时间和窗口大小时，应该考虑到延迟情况。**
3. **如果 LSROSC = 0（可由软件停止振荡），则无论内部振荡模式寄存器（RCM）的第 0 位（LSRSTOP）如何设置，在 HALT 和 STOP 模式下不向看门狗定时器提供计数时钟。**
   当 8 位定时器 H1 使用内部低速振荡时钟时，即使在 HALT/STOP 模式下，也会向 8 位定时器 H1 提供计数时钟。
4. **请确保将第 7 位清除为 0。**

备注
1. f_{RL}： 内部低速振荡时钟频率
2. （）： f_{RL} = 274 kHz（最大值）

图 **26-1.** 选项字节的格式（**2/2**）

地址： 0081H/1081H 注s 1, 2

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	POCMODE

POCMODE	POC 模式选择
0	1.59 V POC 模式（默认）
1	2.7 V/1.59 V POC 模式

注 **1.** POCMODE 只能通过专用的 Flash 存储器编程器写入。在自编程或是自编程的引导交换操作中不能被设置（此时，设置为 1.59 V POC 模式（默认））。但是，因为在引导交换过程中 1081H 的值被拷贝到 0081H，所以在使用引导交换功能时，推荐将 0081H 和 1081H 设定为相同的值。

2. 要改变 POC 模式的设置，可在在 Flash 存储器批擦除（片擦除）之后将该值再次写入 0081H。在指定 block 的存储器被擦除后，不能修改该设置。

注意事项 请确保将第 7 位至第 1 位清除为 0。

地址： 0082H/1082H, 0083H/1083H 注

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0

注 请确保将 0082H 与 0083H 设置为 00H，因为这些地址是保留区域。当使用引导交换时，因为 0082H 与 0083H 要和 1082H 与 1083H 交换，因此也要将 1082 与 1083H 设置为 00H。

地址： 0084H/1084H 注

7	6	5	4	3	2	1	0
0	0	0	0	0	0	OCDEN1	OCDEN0

OCDEN1	OCDEN0	片上调试操作控制
0	0	禁止操作
0	1	禁止设置
1	0	使能操作。在片上调试安全 ID 的验证失效的情况下，不擦除 Flash 存储器的数据。
1	1	使能操作。在片上调试安全 ID 的验证失效的情况下，擦除 Flash 存储器的数据。

注 要片上调试功能，需要将 0084H 设置为 02H 或 03H。因为在引导交换过程中 0084H 和 1084H 被交换，因此也要对 1084H 设置和 0084H 相同的值。

备注 片上调试安全 ID 的详细信息，参见**第二十八章 片上调试功能**

以下是设置选项字节的软件描述示例。

```
OPT       CSEG   AT 0080H
OPTION:   DB     30H          ; 使能看门狗定时器操作（对非法访问检测的操作），
                              ; 看门狗定时器窗口打开时期：50%,
                              ; 看门狗定时器的溢出时间：2^10/fRL,
                              ; 内部低速振荡器可由软件停止
          DB     00H          ; 1.59 V POC 模式
          DB     00H          ; 保留区域
          DB     00H          ; 保留区域
          DB     00H          ; 禁止片上调试操作
```

备注　复位处理期间执行对选项字节的参考。复位处理的时序，参见**第二十三章 复位功能**。

第二十七章　FLASH存储器

78K0/LE3 内置有 Flash 存储器，当安装在电路板上时可以写入、擦除和覆盖。

27.1 内部存储器容量切换寄存器

使用内部存储器容量切换寄存器（IMS）可以选择内部存储器容量。
可以使用 8 位存储器操作指令来设置 IMS。
复位信号的产生会将 IMS 设置为 CFH。

注意事项　复位释放后必须按照表 27-1 对每个产品进行设置。

图 27-1. 内部存储器容量切换寄存器（IMS）的格式

地址：　FFF0H　复位后：　CFH　R/W

符号	7	6	5	4	3	2	1	0
IMS	RAM2	RAM1	RAM0	0	ROM3	ROM2	ROM1	ROM0

RAM2	RAM1	RAM0	内部高速 RAM 容量选择
0	0	0	768 字节
1	1	0	1028 字节
其它情况			禁止设置

ROM3	ROM2	ROM1	ROM0	内部 ROM 容量选择
0	1	0	0	16 KB
0	1	1	0	28 KB
1	0	0	0	32 KB
1	1	0	0	48 KB
1	1	1	1	60 KB
其它情况				禁止设置

表 27-1. 内部存储器容量切换寄存器的设置

Flash 存储器版本（78K0/LE3）	IMS 设置
μPD78F0441，78F0451，78F0461	04H
μPD78F0442，78F0452，78F0462	C6H
μPD78F0443，78F0453，78F0463	C8H
μPD78F0444，78F0454，78F0464	CCH
μPD78F0445，78F0455，78F0465	CFH

27.2 内部扩展RAM 容量切换寄存器

通过内部扩展 RAM 容量切换寄存器（IXS）可以选择内部扩展 RAM 容量。
可以使用 8 位存储器操作指令来设置 IXS。
复位信号的产生会将 IMS 设置为 0CH。

注意事项　复位释放后必须按照表 27-2 对每个产品进行设置。

图 **27-2.** 内部扩展 **RAM** 容量切换寄存器（**IXS**）的格式

地址： FFF4H　复位后： 0CH　R/W

符号	7	6	5	4	3	2	1	0
IXS	0	0	0	IXRAM4	IXRAM3	IXRAM2	IXRAM1	IXRAM0

IXRAM4	IXRAM3	IXRAM2	IXRAM1	IXRAM0	内部扩展 RAM 容量选择
0	1	1	0	0	0 字节
0	1	0	1	0	1024 字节
其它情况					禁止设定

表 **27-2.** 内部扩展 **RAM** 容量切换寄存器设定

Flash 存储器版本（78K0/LE3）	IXS 设置
μPD78F0441，78F0451，78F0461	0CH
μPD78F0442，78F0452，78F0462	
μPD78F0443，78F0453，78F0463	
μPD78F0444，78F0454，78F0464	0AH
μPD78F0445，78F0455，78F0465	

27.3 用Flash存储器编程器写入

使用专用 Flash 存储器编程器，可以在线写入，也可以离线写入 Flash 存储器。

（1）　在线编程

当 78K0/LE3 已经被安装到目标系统上之后，可以重写 Flash 存储器的内容。连接专用 Flash 存储器编程器的连接座必须安装到目标系统。

（2）　离线编程

在 78K0/LE3 被安装到目标系统之前，可以使用专用程序适配器（FA 系列）将数据写入 Flash 存储器。

备注　　FA 系列是 Naito Densei Machida Mfg. Co.，Ltd.的产品。

表 27-3. 78K0/LE3 和专用 Flash 存储器编程器之间的连线

专用 Flash 存储器编程器的引脚配置			使用 CSI10		使用 UART6	
信号名称	I/O	引脚功能	引脚名称	引脚编号	引脚名称	引脚编号
SI/RxD	输入	接收信号	SO10/TxD0 /<TxD6>/P13	62	TxD6/SEG14/P112	28
SO/TxD	输出	发送信号	SI10/RxD0 /<RxD6>/P12	63	RxD6/SEG15/P113	27
SCK	输出	发送时钟	$\overline{\text{SCK10}}$/P11	64	–	–
CLK	输出	至 78K0/LE3 的时钟	–注 1	–	**注 2**	**注 2**
/RESET	输出	复位信号	$\overline{\text{RESET}}$	10	$\overline{\text{RESET}}$	10
FLMD0	输出	模式信号	FLMD0	13	FLMD0	13
V_{DD}	I/O	V_{DD} 电压产生/电压检测	V_{DD}	18	V_{DD}	18
			V_{DD} 注 3	47	V_{DD} 注 3	47
			AV_{REF} 注 4		AV_{REF} 注 4	
GND	–	地	V_{SS}	17	V_{SS}	17
			V_{SS} 注 3	48	V_{SS} 注 3	48
			AV_{SS} 注 4		AV_{SS} 注 4	

注　**1.** 使用 CSI10 的时候，只能使用内部高速振荡时钟（f_{RH}）。

<R>　**2.** 使用 UART6 时，只能使用 X1 时钟（f_X）、外部主系统时钟（f_{EXCLK}）或内部高速振荡时钟（f_{RH}）。使用 Flash 存储器编程器的时钟输出时，将 PG-FP5 或 FL-PR5 的 CLK 连接到 EXCLK/X2/P122（引脚编号 14）。

3. 仅限μPD78F044x。

4. 仅限μPD78F045X 和 78F046x。

<R>　注意事项　**当通过 Flash 存储器编程器写入时，只有底侧引脚（引脚编号 27 和 28）对应 UART6 引脚（RxD6 和 TxD6）。不能通过顶侧引脚（引脚编号 63 和 62）执行写入。**

使用适配器对 Flash 存储器进行写入时，推荐连接的示例。

图 27-3. 在 3 线串行 I/O（CSI10）模式下使用适配器对 Flash 存储器进行写入的连线示例

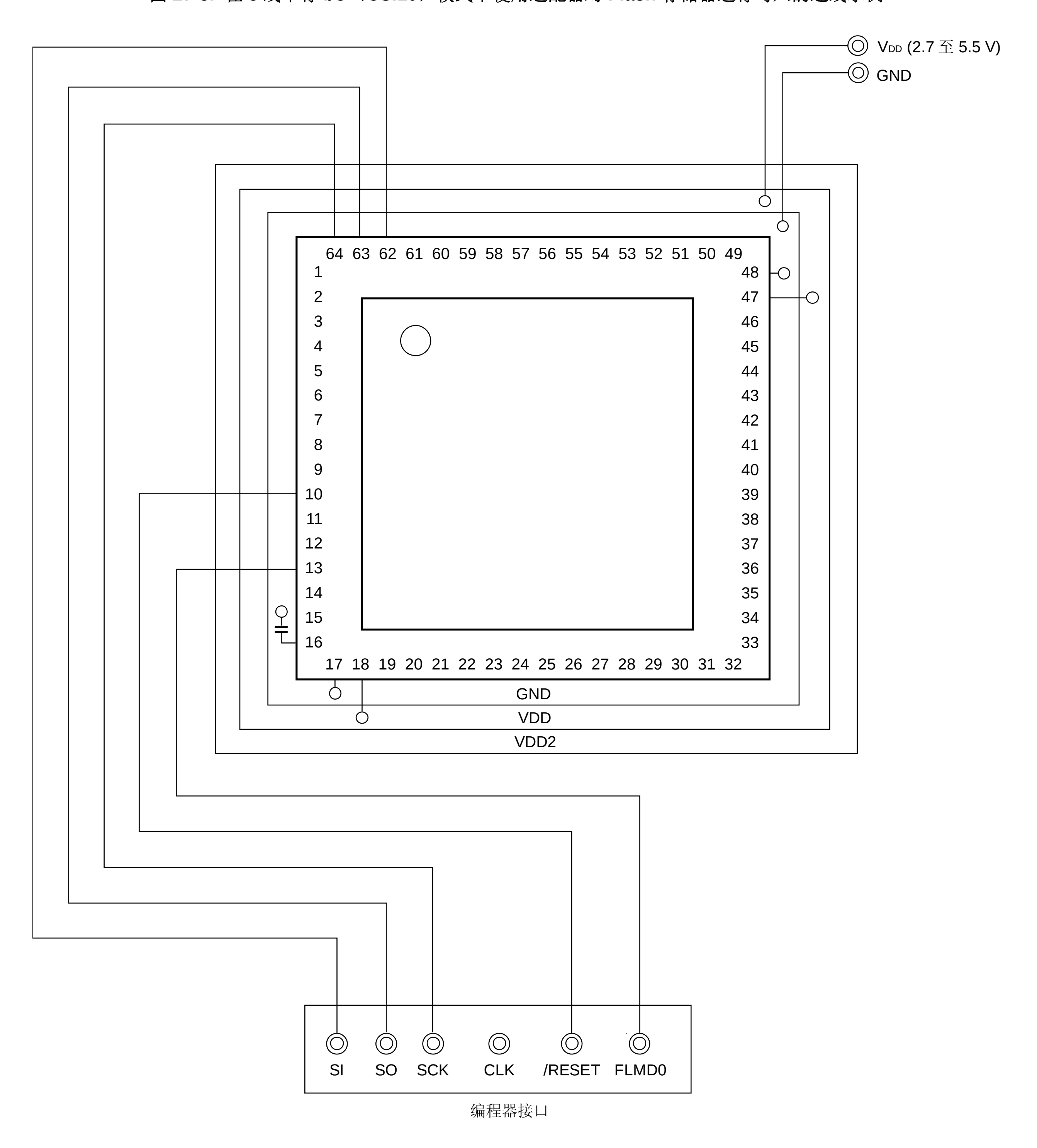
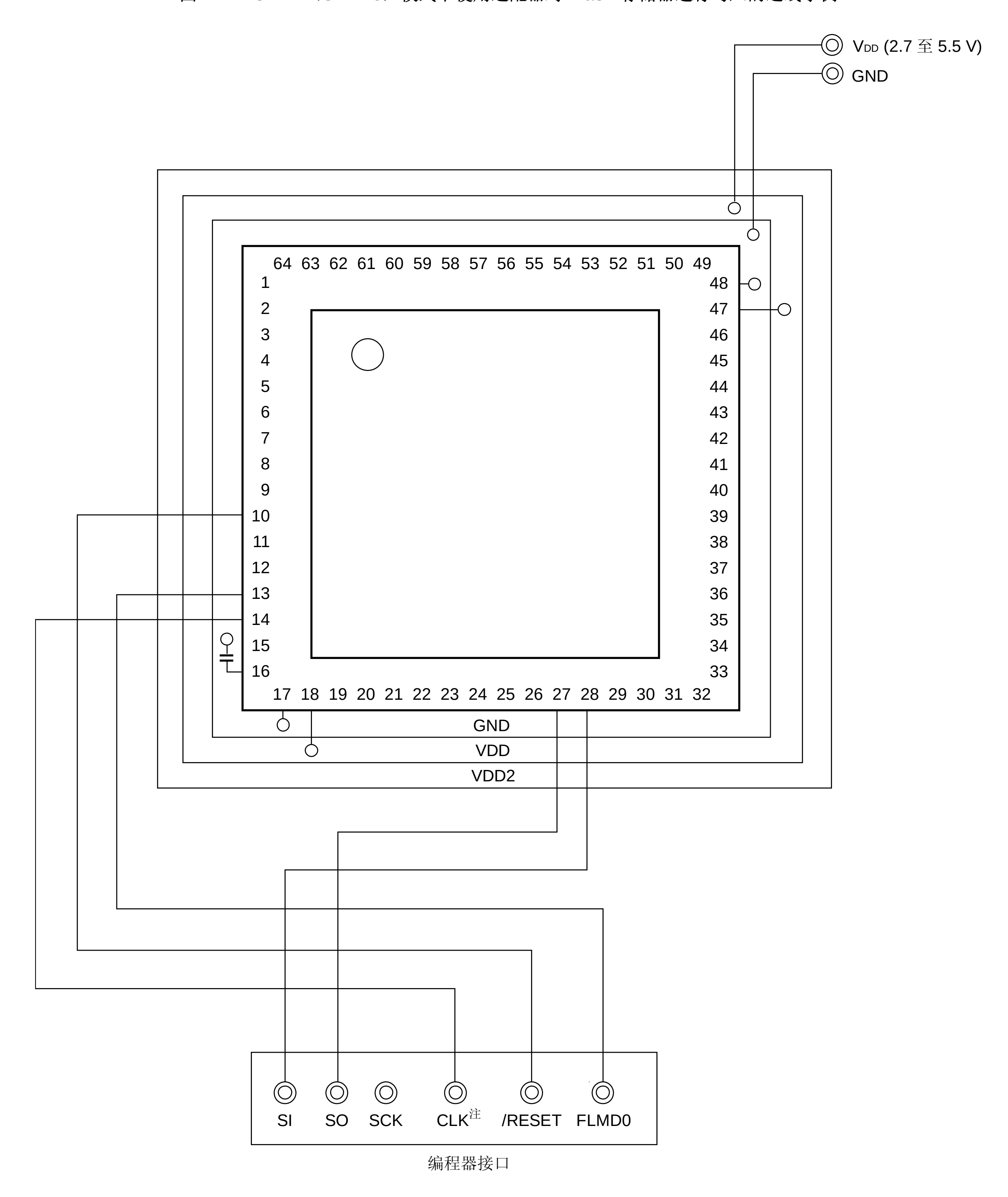


图 27-4. UART（UART6）模式下使用适配器对 Flash 存储器进行写入的连线示例

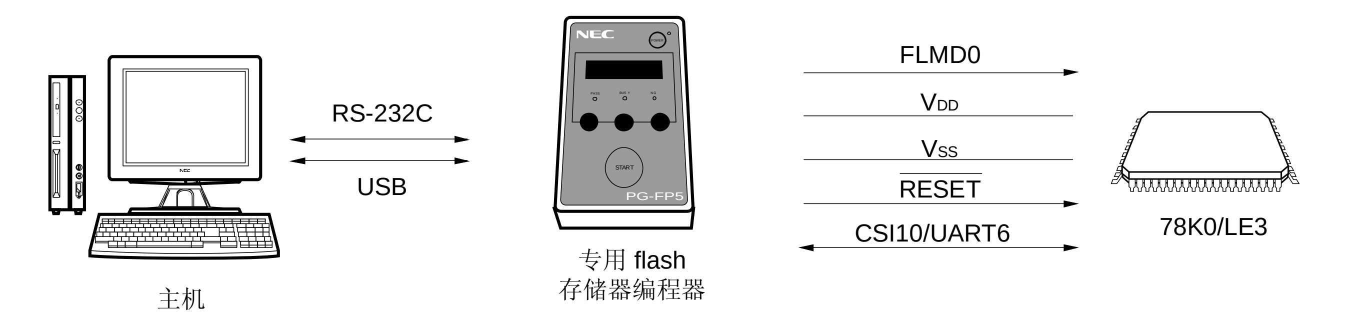


注　　上图展示了当使用 PG-FP5 或 FL-PR5 输出时钟的连线示例。

27.4 编程环境

将程序写入 78K0/LE3 Flash 存储器所需的编程环境如下所示。

图 27-5. 向 Flash 存储器写入程序的环境



必需有一个主机来控制专用 Flash 存储器编程器。

专用 Flash 存储器编程器与 78K0/LE3 之间的接口，使用 CSI10 或 UART6 来操作比如写入和擦除。要离线写入 Flash 存储器，必需一个专用程序适配器（FA 系列）。

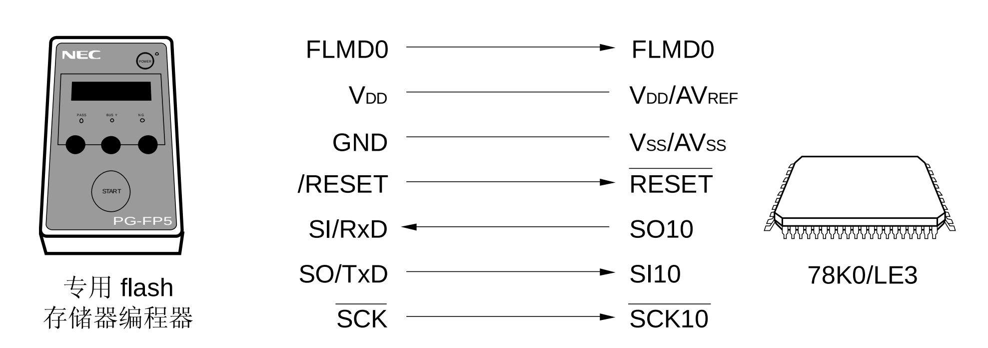
27.5 通信模式

通过 78K0/LE3 的 CSI10 或 UART6，借助串行通信在专用 Flash 存储器编程器与 78K0/LE3 之间建立通信。

（1） CSI10

传输速率： 2.4 kHz 至 2.5 MHz

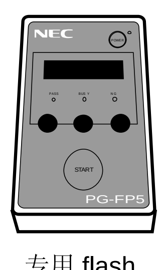
图 27-6. 和专用 Flash 存储器编程器的通信（CSI10）

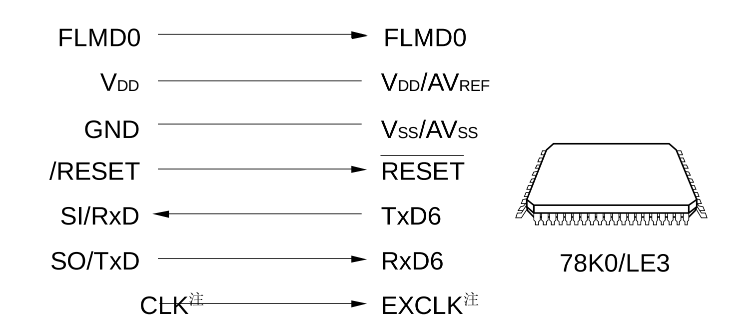


（2） **UART6**

传输速率： 115200 bps

图 **27-7.** 和专用 **Flash** 存储器编程器的通信（**UART6**）





专用 flash
存储器编程器

注　上图展示了当使用 PG-FP5 或 FL-PR5 输出时钟的连线示例。

注意事项　当通过 Flash 存储器编程器写入时，只有底侧引脚（引脚编号 27 和 28）对应 UART6 引脚（RxD6 和 TxD6）。不能通过顶侧引脚（引脚编号 63 和 62）执行写入。

专用 Flash 存储器编程器为 78K0/LE3 产生下列信号。详细信息，参考 PG-FP5 或 FL-PR5 的用户手册。

表 **27-4.** 引脚连接

专用 Flash 存储器编程器			78K0/LE3	连接	
信号名称	I/O	引脚功能	引脚名称	CSI10	UART6
FLMD0	输出	模式信号	FLMD0	◎	◎
V_{DD}	I/O	V_{DD} 电压产生/电源监测	V_{DD}，AV_{REF} 注 3	◎	◎
GND	–	地	V_{SS}，AV_{SS} 注 3	◎	◎
CLK	输出	时钟输出到 78K0/LE3	注 1	× 注 2	○ 注 1
/RESET	输出	复位信号	$\overline{\text{RESET}}$	◎	◎
SI/RxD	输入	接收信号	SO10 或 TxD6	◎	◎
SO/TxD	输出	发送信号	SI10 或 RxD6	◎	◎
SCK	输出	传输时钟	$\overline{\text{SCK10}}$	◎	×

<R> 注 **1.** 使用 UART6 时，只能使用 X1 时钟（fX）、外部主系统时钟（f_{EXCLK}）或内部高速振荡时钟（f_{RH}）。使用 Flash 存储器编程器的时钟输出时，将 PG-FP5 或 FL-PR5 的 CLK 连接到 EXCLK/X2/P122（引脚编号 18）。

2. 使用 CSI10 时，只能选择内部高速振荡时钟（f_{RH}）。

3. 仅限 μPD78F045x 和 78F046x。

备注　◎：　请确保连接引脚。

○：　如果由目标板产生该信号，则该引脚不需要连接。

×：　该引脚不需要连接。

27.6 在线方式的引脚连接

要对 Flash 存储器进行在线写入操作时，目标系统必须提供连接专用 Flash 存储器编程器的连接座。电路板上首先要提供一个选择功能，可以选择正常操作模式或 Flash 存储器编程模式。

当设置为 Flash 存储器编程模式时，那些 Flash 存储器编程不使用的引脚状态与复位后的瞬时状态相同。因此如果外部设备不能识别复位后的瞬时状态，则引脚必须按照下列描述来处理。

27.6.1 FLMD0 引脚

在正常操作模式下，FLMD0 引脚的输入电压为 0V。在 Flash 存储器编程模式中，V_{DD} 写入电压被提供给 FLMD0 引脚。FLMD0 引脚的连接示例如下所示。

图 27-8. FLMD0 引脚连接示例

27.6.2 串行接口引脚

各个串行接口使用的引脚如下所示。

表 27-5. 各个串行接口使用的引脚

串行接口	使用引脚
CSI10	SO10，SI10，$\overline{\text{SCK10}}$
UART6	TxD6，RxD6

要将专用 Flash 存储器编程器和串行接口的引脚相连时，该串行接口已经连接到板上其它设备，则必须特别注意，使得信号之间不冲突，或者其他设备不出现误操作故障。

（1） 信号冲突

要将专用 Flash 存储器编程器（输出）和串行接口的引脚（输入）相连时，该串行接口已经连接到板上其它设备（输出），则会产生信号冲突。为了避免这种冲突，应隔离与其他设备的连接，或者使另一个设备进入输出高阻抗状态。

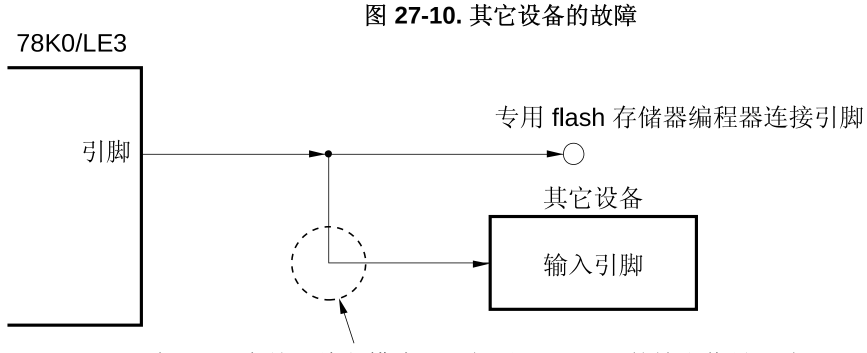
图 27-9. 信号冲突（串行接口的输入引脚）

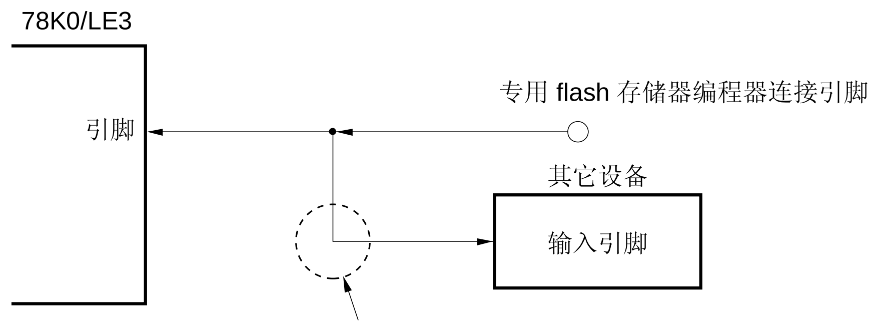
78K0/LE3
输入引脚
信号冲突
专用 flash 存储器编程器连接引脚
其它设备
输出引脚
在 flash存储器编程模式下，设备的输出信号与专用 flash 编程器
发送的信号发生冲突，因此必须隔离该设备的信号。

（2） 其它设备的故障

如果专用 Flash 存储器编程器（输出或输入）和串行接口的引脚（输入或输出）相连时，该串行接口已经连接到板上其它设备（输入），则信号可能会输出到另一个设备，从而引起该设备的故障。为了避免这种情况，应隔离与该设备的连接。

图 27-10. 其它设备的故障



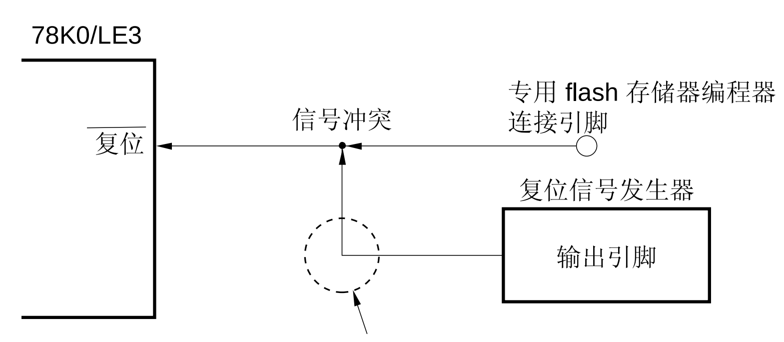


27.6.3 $\overline{\text{RESET}}$ 引脚

如果将专用 Flash 存储器编程器的复位信号连接到 $\overline{\text{RESET}}$ 引脚，该引脚已经连接到板上的复位信号发生器，则会发生信号冲突。为了避免这种情况，应隔离与复位信号发生器的连接。

设置为 Flash 存储器编程模式时，如果从用户系统输入复位信号，则 Flash 存储器不能进行正确编程。因此除了专用 Flash 存储器编程器的复位信号外，不要输入任何信号。

图 27-11. 信号冲突（$\overline{\text{RESET}}$ 引脚）



27.6.4 端口引脚

当设置为 Flash 存储器编程模式时，那些 Flash 存储器编程不使用的引脚状态与复位后的瞬时状态相同。因此，如果外部设备（与端口相连）不能识别复位后的瞬时状态，则必须通过一个电阻将端口引脚连接到 V_{DD} 或 V_{SS}。

27.6.5 REGC引脚

与正常操作的方式相同，通过一个电容（0.47 至 1 μF：推荐）将 REGC 引脚连接到 GND。

27.6.6 其它信号引脚

使用在线时钟时，X1 和 X2 的连接与正常操作模式下的状态相同。

要从编程器输入操作时钟，但是，将 PG-FP5 或 FL-PR5 的 CLK 连接到 EXCLK/X2/P122。

注意事项　1. 使用 CSI10 的时候，只能使用内部高速振荡时钟（f_{RH}）。

<R> **2. 使用 UART6 时，可以使用 X1 时钟（f_X）、外部主系统时钟（f_{EXCLK}）或内部高速振荡时钟（f_{RH}）。**

27.6.7 电源

要使用 Flash 存储器编程器的供电电压输出，则将 V_{DD} 引脚与 Flash 存储器编程器的 V_{DD} 连接，将 V_{SS} 引脚与 Flash 存储器编程器的 GND 连接。

要使用在线供电电压，连接要符合正常操作模式。

但是，如果要使用 Flash 存储器编程器的电源监控功能，必须将 V_{DD} 和 V_{SS} 引脚分别与 Flash 存储器编程器的 V_{DD} 和 GND 相连接。

其它供电电压（AV_{REF} 和 AV_{SS}）与正常操作模式相同。

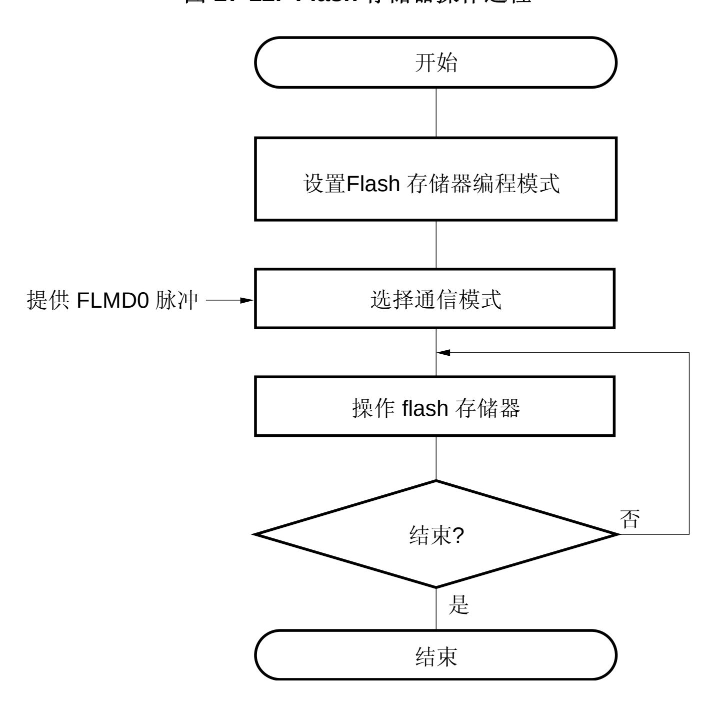
27.7 编程方法

27.7.1 控制Flash 存储器

下图展示了操作 Flash 存储器的过程。

<R>

图 27-12. Flash 存储器操作过程



27.7.2 Flash存储器编程模式

如果通过专用 Flash 存储器编程器重写 Flash 存储器的内容，则必须将 78K0/LE3 设置为 Flash 存储器编程模式。要设置该模式，必须将 FLMD0 引脚连接到 V_{DD}，并清除复位信号。

当在线写入 Flash 存储器时，使用跳线改变模式。

图 27-13. Flash 存储器编程模式

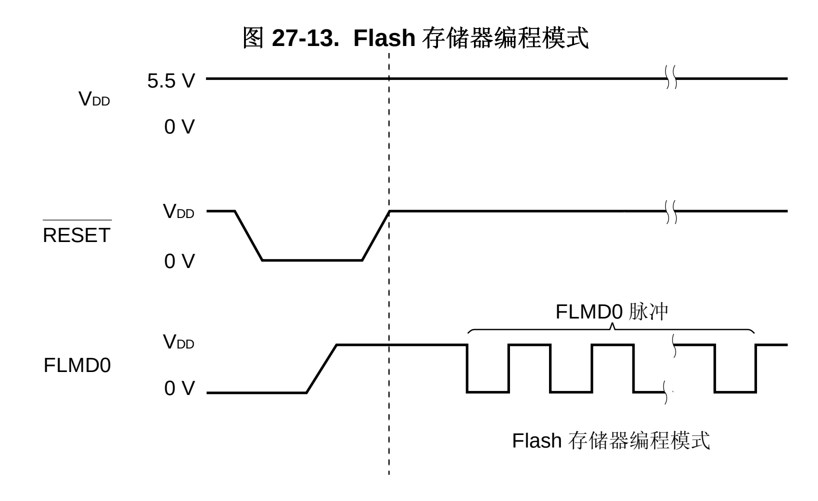


表 27-6. 复位释放后 FLMD0 引脚与操作模式之间的关系

FLMD0	操作模式
0	正常操作模式
V_{DD}	Flash 存储器编程模式

27.7.3 选择通信的模式

在 78K0/LE3 中，进入专用 Flash 存储器编程模式后，通过输入到 FLMD0 引脚的脉冲来选择通信模式。这些 FLMD0 脉冲由 Flash 存储器编程器产生。

下表展示了脉冲数量与通信模式之间的关系。

表 27-7. 通信模式

<R>

通信模式	标准设置[注 1]				使用引脚	外围时钟	FLMD0 脉冲的数量
	端口	速度	频率	倍率			
UART（UART6）	UART-Ext-OSC	115，200 bps[注 3]	2 M 至 10 MHz[注 2]	1.0	TxD6，RxD6	f_X	0
	UART-Ext-FP5CLK					f_{EXCLK}	3
	UART-内部-OSC		–			f_{RH}	5
3 线串行 I/O（CSI10）	CSI-内部-OSC	2.4 kHz 至 2.5 MHz	–		SO10，$\overline{\text{SI10}}$，SCK10	f_{RH}	8

注　**1.** 编程器 GUI 的标准设置选择项目。

2. 可以设置的范围取决于电压而不同。详情参见**第三十章 电气特性（标准产品）**。

3. 因为波特率误差以外的因素，比如信号波形瞬变，也会影响 UART 通信，因此必须象测量波特率误差一样全面评估瞬变。

注意事项　当选择 UART6 时，在接收到 FLMD0 脉冲后，基于专用 Flash 存储器编程器发出的复位命令来计算接收时钟。

备注　f_X：　X1 时钟

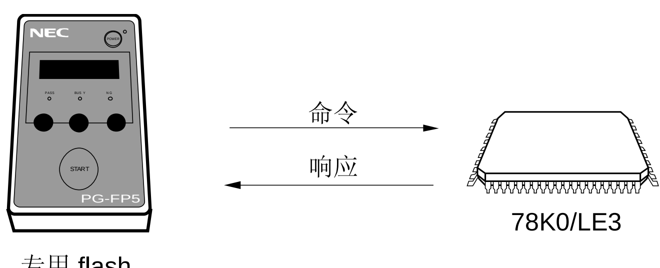
f_{EXCLK}：　外部主系统时钟

f_{RH}：　内部高速振荡时钟

27.7.4 通信命令

78K0/LE3 使用命令与专用 Flash 存储器编程器进行通信。从 Flash 存储器编程器发往 78K0/LE3 的信号称为命令，从 78K0/LE3 发往专用 Flash 存储器编程器的命令称为响应。

图 27-14. 通信命令



78K0/LE3 的 Flash 存储器控制命令列表如下。所有这些命令都由编程器发出，78K0/LE3 根据各个命令执行对应的处理。

表 27-8. Flash 存储器控制命令

类别	命令名称	功能
校验	校验	比较 flash 存储器指定区域的内容和编程器发送的数据。
擦除	片擦除	擦除整个 flash 存储器。
	块擦除	擦除 flash 存储器的指定区域。
空白检测	块空白检测	检查 flash 存储器中一个指定的块是否被正确擦除。
数据写	编程	将数据写入 flash 存储器的指定区域。
获取信息	状态	获取当前操作状态（状态数据）。
	硅标记	获取 78K0/Lx3 版本和固件版本信息。
	获取版本	获取 78K0/Lx3 版本和固件版本。
	校验和	获取指定区域的数据的校验和。
安全设置	安全设置	设置安全信息。
其它	复位	用来检测通信的同步状态。
	振荡频率设置	指定振荡频率

78K0/LE3 根据专用 Flash 存储器编程器发出的命令返回一个响应。78K0/LE3 发出的响应名称列表如下。

表 27-9. 响应命令

响应名称	功能
ACK	响应命令/数据
NAK	响应非法命令/数据

27.8 安全性设置

78K0/LE3 支持安全功能，可以禁止重写已经写入内部 flash 存储器中的用户程序，因此未授权用户就不能更改程序。

以下的操作是使用安全设定命令能够执行的操作。在下一次设置编程模式之前，安全性设置有效。

- 禁止批量擦除（片擦除）
  在线/离线编程期间使用该设置，禁止对 Flash 存储器的所有块执行块擦除命令和批量擦除（片擦除）命令。一旦批量擦除（片擦除）命令的执行被禁止，所有禁止设置（包括批量擦除（片擦除）的禁止）将不能被取消。

  注意事项 在设置了批量擦除的安全性设置之后，不能再对该设备执行擦除操作。此外，因为擦除命令被禁止，即使执行写入命令，如果数据与已经写到 Flash 存储器中的数据不同，也不能被写入。

- 禁止块擦除
  在线/离线编程期间使用该设置，禁止对 Flash 存储器的指定块执行块擦除命令。但是，可以通过自编程的方式来擦除块。

- 禁止写入
  在线/离线编程期间使用该设置，禁止对 Flash 存储器的所有块执行写入命令和块擦除命令。但是，可以通过自编程的方式来擦除块。

- 禁止重写引导簇 0
  该设置禁止对 Flash 存储器引导簇 0（0000H 至 0FFFH）执行批量擦除（片擦除）命令、块擦除命令和写命令。

  注意事项 如果对重写引导簇 0 已经进行了安全性设置，则不能再重写该设备的引导簇 0。

Flash 存储器出库时的默认设置，批量擦除（片擦除）、块擦除、写入命令和重写引导簇 0 都被使能。以上安全性设置可以通过在线/离线编程和自编程来设置。各项安全性设置可以组合使用。

通过执行批量擦除（片擦除）命令，擦除块和写入的禁止被清除。

表 27-10 展示了 78K0/LE3 安全功能使能时擦除命令与写入命令之间的关系。

表 27-10. 安全功能使能和命令之间的关系

（1） 在线/离线编程期间

有效的安全功能	执行的命令		
	批量擦除（片擦除）	块擦除	写入
禁止批量擦除（片擦除）	不能批量擦除	块不能擦除。	不能执行[注]。
禁止块擦除	可以批量擦除		可以执行。
禁止写入			不能执行。
禁止重写引导簇 0	不能批量擦除	引导簇 0 不能擦除。	引导簇 0 不能写入。

注 确认没有数据被写入到写入区域。因为批量擦除（片擦除）命令被禁止后，数据不能被擦除，如果数据没有被擦除，则不要写入数据。

（2） 自编程期间

有效的安全功能	执行的命令	
	块擦除	写入
禁止批量擦除（片擦除）	块可以擦除。	可以执行。
禁止块擦除		
禁止写入		
禁止引导簇 0 重写	引导簇 0 不能擦除。	引导簇 0 不能写入。

表 27-11 展示了在各种编程模式下如何执行安全设置。

表 27-11. 各种编程模式下的安全设置

（1） 在线/离线编程

安全功能	安全设置	如何禁止安全设置
禁止批量擦除（片擦除）	通过专用 Flash 存储器编程器的 GUI 等进行设置。	设置之后不能禁止。
禁止块擦除		执行批量擦除（片擦除）命令。
禁止写入		
禁止引导簇 0 重写		设置之后不能禁止。

（2） 自编程

安全功能	安全设置	怎样禁止安全设置
禁止批量擦除（片擦除）	使用信息库进行设置。	设置之后不能禁止。
禁止块擦除		在线/离线 编程期间执行批量擦除（片擦除）命令（自编程期间不能禁止）。
禁止写入		
禁止引导簇 0 重写		设置之后不能禁止。

<R> 27.9 使用PG-FP5 时各个命令的处理时间（参考）

当使用 PG-FP5 作为专用 flash 存储器编程器时，下表列出了各个命令的处理时间（参考）。

表 27-12. 使用 PG-FP5 时各个命令的处理时间（参考）（1/3）

（1）μPD78F0441，78F0451，78F0461（产品的内部 ROM：16 KB）

PG-FP5 的命令	端口：CSI-内部-OSC（内部高速振荡时钟（f_{RH}））速度：2.5 MHz	端口：UART-内部-OSC（内部高速振荡时钟（f_{RH}））速度：115200 bps	端口：UART-Ext-OSC（X1 时钟（f_X））速度：115200 bps		端口：UART-Ext-FP5CLK（外部主系统时钟（f_{EXCLK}）），速度：115200 bps	
			频率：2.0 MHz	频率：10 MHz	频率：2.0 MHz	频率：10 MHz
特征签名	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）
空白检测	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）
擦除	1.5 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）
编程	3 s（典型值）	4 s（典型值）	4 s（典型值）	4 s（典型值）	4 s（典型值）	4 s（典型值）
校验	2 s（典型值）	3 s（典型值）	3 s（典型值）	3 s（典型值）	3 s（典型值）	3 s（典型值）
E.P.V	3.5 s（典型值）	4 s（典型值）	4 s（典型值）	4 s（典型值）	4 s（典型值）	4 s（典型值）
校验和	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）
安全	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）

（2）μPD78F0442，78F0452，78F0462（产品的内部 ROM：24 KB）

PG-FP5 的命令	端口：CSI-内部-OSC（内部高速振荡时钟（f_{RH}））速度：2.5 MHz	端口：UART-内部-OSC（内部高速振荡时钟（f_{RH}））速度：115200 bps	端口：UART-Ext-OSC（X1 时钟（f_X））速度：115200 bps		端口：UART-Ext-FP5CLK（外部主系统时钟（f_{EXCLK}））速度：115200 bps	
			频率：2.0 MHz	频率：10 MHz	频率：2.0 MHz	频率：10 MHz
特征签名	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）
空白检测	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）
擦除	1.5 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）
编程	4 s（典型值）	5.5 s（典型值）	5.5 s（典型值）	5.5 s（典型值）	5.5 s（典型值）	5.5 s（典型值）
校验	2.5 s（典型值）	4 s（典型值）	4 s（典型值）	4 s（典型值）	4 s（典型值）	4 s（典型值）
E.P.V	4.5 s（典型值）	5.5 s（典型值）	5.5 s（典型值）	5.5 s（典型值）	5.5 s（典型值）	5.5 s（典型值）
校验和	1.5 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）
安全	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）

注意事项　当执行引导交换时，不要在专用 **flash** 存储器编程器上使用 **E.P.V.**命令。

表 27-12. 使用 PG-FP5 时各个命令的处理时间（参考）（2/3）

（3）μPD78F0443，78F0453，78F0463（产品的内部 ROM： 32 KB）

PG-FP5 的命令	端口：CSI-内部-OSC（内部高速振荡时钟（f_{RH}））速度： 2.5 MHz	端口：UART-内部-OSC（内部高速振荡时钟（f_{RH}））速度： 115200 bps	端口：UART-Ext-OSC（X1 时钟（f_X））速度：115200 bps		端口：UART-Ext-FP5CLK（外部主系统时钟（f_{EXCLK}））速度：115200 bps	
			频率：2.0 MHz	频率：10 MHz	频率：2.0 MHz	频率：10 MHz
特征签名	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）
空白检测	1.5 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）
擦除	1.5 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）
编程	4.5 s（典型值）	6.5 s（典型值）	6.5 s（典型值）	6.5 s（典型值）	6.5 s（典型值）	6.5 s（典型值）
校验	2.5 s（典型值）	5 s（典型值）	5 s（典型值）	5 s（典型值）	5 s（典型值）	5 s（典型值）
E.P.V	5.5 s（典型值）	7 s（典型值）	7 s（典型值）	7 s（典型值）	7 s（典型值）	7 s（典型值）
校验和	1.5 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）
安全	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）

（4）μPD78F0444，78F0454，78F0464（产品的内部 ROM： 48 KB）

PG-FP5 的命令	端口：CSI-内部-OSC（内部高速振荡时钟（f_{RH}））速度： 2.5 MHz	端口：UART-内部-OSC（内部高速振荡时钟（f_{RH}））速度： 115200 bps	端口：UART-Ext-OSC（X1 时钟（f_X））速度：115200 bps		端口：UART-Ext-FP5CLK（外部主系统时钟（f_{EXCLK}））速度：115200 bps	
			频率：2.0 MHz	频率：10 MHz	频率：2.0 MHz	频率：10 MHz
特征签名	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）
空白检测	1 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）
擦除	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）
编程	6.5 s（典型值）	9.5 s（典型值）	9.5 s（典型值）	9.5 s（典型值）	9.5 s（典型值）	9.5 s（典型值）
校验	3.5 s（典型值）	7 s（典型值）	7 s（典型值）	7 s（典型值）	7 s（典型值）	7 s（典型值）
E.P.V	7.5 s（典型值）	10 s（典型值）	10 s（典型值）	10 s（典型值）	10 s（典型值）	10 s（典型值）
校验和	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）
安全	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）

注意事项　当执行引导交换时，不要在专用 **flash** 存储器编程器上使用 **E.P.V.**命令。

表 27-12. 使用 PG-FP5 时各个命令的处理时间（参考）（3/3）

（5）μPD78F0445，78F0455，78F0465（产品的内部 ROM： 60 KB）

PG-FP5 的命令	端口：CSI-内部-OSC（内部高速振荡时钟（f_{RH}））速度： 2.5 MHz	端口：UART-内部-OSC（内部高速振荡时钟（f_{RH}））速度： 115200 bps	端口：UART-Ext-OSC（X1 时钟（f_X））速度：115200 bps		端口：UART-Ext-FP5CLK（外部主系统时钟（f_{EXCLK}））速度：115200 bps	
			频率：2.0 MHz	频率：10 MHz	频率：2.0 MHz	频率：10 MHz
特征签名	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）
空白检测	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）
擦除	2 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）
编程	8 s（典型值）	12 s（典型值）	12 s（典型值）	11.5 s（典型值）	12 s（典型值）	11.5 s（典型值）
校验	4.5 s（典型值）	8.5 s（典型值）	8.5 s（典型值）	8.5 s（典型值）	8.5 s（典型值）	8.5 s（典型值）
E.P.V	9 s（典型值）	12.5 s（典型值）	12.5 s（典型值）	12.5 s（典型值）	12.5 s（典型值）	12.5 s（典型值）
校验和	2 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）	1.5 s（典型值）
安全	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）	1 s（典型值）

注意事项　当执行引导交换时，不要在专用 **flash** 存储器编程器上使用 **E.P.V.**命令。

<R>

27.10 通过自编程进行Flash存储器编程

78K0/LE3 微控制器支持自编程功能，可以通过用户程序重写 Flash 存储器。因为该功能允许用户使用自编程库重写 flash 存储器，因此可用于程序的现场升级。

如果自编程期间发生中断，可以暂时停止自编程操作，执行中断服务程序。要执行中断服务程序，应在停止自编程操作后，恢复正常操作模式，并执行 EI 指令。而在之后恢复自编程模式时，可以继续自编程操作。

注意事项 **1.** 当 **CPU** 运行于副系统时钟时，不能使用自编程功能。

2. 在自编程期间，内部高速振荡器的振荡开始，而无论 **RSTOP** 标志如何设置（内部振荡模式寄存器（**RCM**）的第 **0** 位）。即使执行 **STOP** 指令，内部高速振荡器的振荡也不会停止。

3. 自编程期间，**FLMD0** 引脚应输入高电平。

4. 在启动自编程之前，请确保执行 **DI** 指令。
自编程功能检查中断请求标志（**IF0L**，**IF0H**，**IF1L** 和 **IF1H**）。如果产生中断请求，则停止自编程操作。

5. 即使在 **DI** 状态下，未屏蔽的中断请求也会停止自编程。要避免这种情况，可以使用中断屏蔽标志寄存器（**MK0L**，**MK0H**，**MK1L** 和 **MK1H**）屏蔽该中断。

6. 自编程的入口程序被分配在 **0000H** 至 **7FFFH**。

图 **27-15.** 自编程的操作模式和存储器映射（μ**PD78F0445**）

FFFFH
FF00H
FEFFH
FB00H
FAFFH
F800H
F7FFH
F400H
F3FFH
F000H
EFFFH
8000H
7FFFH
0000H
SFR
内部高速 RAM
内部扩展 RAM
保留
Flash 存储器
(用户区域)
Flash 存储器
控制固件
ROM
禁止
访问
指令可以从用户区域取出
正常模式
允许
访问
指令可以从用户区域
和固件 ROM取出.
自编程模式

下图展示了使用自编程范例库重写 flash 存储器的流程。

图 27-16. 自编程的流程（重写 **Flash** 存储器）

启动自编程

FLMD0 引脚低电平到高电平

FLASH 起始

设定操作环境

FlashEnv

检测 FLMD

正常完成？

否

是

FlashBlockBlankCheck

已擦除？

否

是

FlashBlockErase

正常完成？

是

Flash 字写入

正常完成？

否

否

是

FlashBlockVerify

正常完成？

否

是

FlashEnd

FLMD0 引脚高电平到低电平

自编程完毕

备注　　自编程库的详情参见 **78K0** 微控制器自编程库类型 **01** 用户手册（**U18274E**）。

下表列出了自编程库所需的处理时间和中断响应时间。

表 27-13. 处理时间和中断响应时间（1/4）
（当正常模型库和入口 RAM 被分配在短直接寻址范围之外）

功能	处理时间（单位：μs）			中断响应
	RSTOP = 0 且 RSTS = 1（在内部高速振荡器稳定操作期间）		RSTOP = 1（内部高速振荡器停止）[注]	
	MCS = 0（内部高速振荡时钟）	MCS = 1（高速系统时钟）	MCS = 1（高速系统时钟）	
自编程启动功能	$34/f_{CPU}$	$34/f_{CPU}$	$34/f_{CPU}$	禁止
自编程结束功能	$34/f_{CPU}$	$34/f_{CPU}$	$34/f_{CPU}$	禁止
初始化功能	$55/f_{CPU}+1140$	$55/f_{CPU}+1140$	$55/f_{CPU}+1912$	禁止
块擦除功能	$179/f_{CPU}+353193$	$179/f_{CPU}+353193$	$179/f_{CPU}+353965$	使能
字写入功能	$333/f_{CPU}+1154+2142\times W$	$333/f_{CPU}+1154+2142\times W$	$333/f_{CPU}+1927+2142\times W$	使能
块校验功能	$179/f_{CPU}+25596$	$179/f_{CPU}+25596$	$179/f_{CPU}+26369$	使能
块空白检测功能	$179/f_{CPU}+12805$	$179/f_{CPU}+12805$	$179/f_{CPU}+13578$	使能
获取信息功能 选项值 03H	$180/f_{CPU}+1065$	$180/f_{CPU}+1065$	$180/f_{CPU}+1838$	禁止
获取信息功能 选项值 04H	$190/f_{CPU}+1056$	$190/f_{CPU}+1056$	$190/f_{CPU}+1829$	禁止
获取信息功能 选项值 05H	$350/f_{CPU}+1041$	$350/f_{CPU}+1041$	$350/f_{CPU}+1813$	禁止
设置信息功能	$80/f_{CPU}+753218$	$80/f_{CPU}+753218$	$80/f_{CPU}+753990$	使能
模式检查功能	$36/f_{CPU}+952$	$36/f_{CPU}+952$	$36/f_{CPU}+1724$	禁止
EEPROM 写入功能	$333/f_{CPU}+1297+2286\times W$	$333/f_{CPU}+1297+2286\times W$	$333/f_{CPU}+2069+2286\times W$	使能

注　这是当自编程启动功能执行后立即执行该功能的功能处理时间。自编程启动功能之外的功能执行之后的处理时间，和 RSTOP = 0 的情况一样。

备注　RSTOP：　内部振荡模式寄存器（RCM）的第 0 位
RSTS：　RCM 的第 7 位
MCS：　主时钟模式寄存器（MCM）的第 1 位
f_{CPU}：　CPU 时钟频率
W：　将要写入的字的数量（1 字 = 4 字节）

表 **27-13.** 处理时间和中断响应（**2/4**）
（当正常模型库和入口 **RAM** 被分配在短直接寻址范围之内）

功能		处理时间（单位：μs）			中断响应
		RSTOP = 0 且 RSTS = 1（在内部高速振荡器稳定操作期间）		RSTOP = 1（内部高速振荡器停止）注	
		MCS = 0（内部高速振荡时钟）	MCS = 1（高速系统时钟）	MCS = 1（高速系统时钟）	
自编程启动功能		34/fCPU	34/fCPU	34/fCPU	禁止
自编程结束功能		34/fCPU	34/fCPU	34/fCPU	禁止
初始化功能		55/fCPU+462	55/fCPU+462	55/fCPU+473	禁止
块擦除功能		179/fCPU+352516	179/fCPU+352516	179/fCPU+352528	使能
字写入功能		333/fCPU+477+2142×W	333/fCPU+477+2142×W	333/fCPU+488+2142×W	使能
块校验功能		179/fCPU+24918	179/fCPU+24918	179/fCPU+24930	使能
块空白检测功能		179/fCPU+12128	179/fCPU+12128	179/fCPU+12139	使能
获取信息功能	选项值 03H	180/fCPU+388	180/fCPU+388	180/fCPU+399	禁止
	选项值 04H	190/fCPU+378	190/fCPU+378	190/fCPU+390	禁止
	选项值 05H	350/fCPU+363	350/fCPU+363	350/fCPU+375	禁止
设置信息功能		80/fCPU+752540	80/fCPU+752540	80/fCPU+753654	使能
模式检查功能		36/fCPU+274	36/fCPU+274	36/fCPU+286	禁止
EEPROM 写入功能		333/fCPU+619+2286×W	333/fCPU+619+2286×W	333/fCPU+630+2286×W	使能

注　这是当自编程启动功能执行后立即执行该功能的功能处理时间。自编程启动功能之外的功能执行之后的处理时间，和 RSTOP = 0 的情况一样。

备注　RSTOP：内部振荡模式寄存器（RCM）的第 0 位
RSTS：RCM 的第 7 位
MCS：主时钟模式寄存器（MCM）的第 1 位
f_{CPU}：CPU 时钟频率
W：将要写入的字的数量（1 字 = 4 字节）

表 27-13. 处理时间和中断响应（3/4）
（当静态模型库和入口 RAM 被分配在短直接寻址范围之外）

功能		处理时间（单位：µs）			中断响应
		RSTOP = 0 且 RSTS = 1 （在内部高速振荡器稳定操作期间）		RSTOP = 1 （内部高速振荡器停止）注	
		MCS = 0 （CPU 运行于内部高速振荡 时钟）	MCS = 1 （CPU 运行于高速系统时钟）	MCS = 1 （CPU 运行于高速系统时钟）	
自编程启动功能		34/f_{CPU}	34/f_{CPU}	34/f_{CPU}	禁止
自编程结束功能		34/f_{CPU}	34/f_{CPU}	34/f_{CPU}	禁止
初始化功能		55/f_{CPU}+1140	55/f_{CPU}+1140	55/f_{CPU}+1912	禁止
块擦除功能		136/f_{CPU}+353193	136/f_{CPU}+353193	136/f_{CPU}+353965	使能
字写入功能		272/f_{CPU}+1154+ 2142×W	272/f_{CPU}+1154+ 2142×W	272/f_{CPU}+1927+ 2142×W	使能
块校验功能		136/f_{CPU}+25596	136/f_{CPU}+25596	136/f_{CPU}+26369	使能
块空白检测功能		136/f_{CPU}+12805	136/f_{CPU}+12805	136/f_{CPU}+13578	使能
获取信息功能	选项值 03H	134/f_{CPU}+1065	134/f_{CPU}+1065	134/f_{CPU}+1838	禁止
	选项值 04H	144/f_{CPU}+1056	144/f_{CPU}+1056	144/f_{CPU}+1829	禁止
	选项值 05H	304/f_{CPU}+1041	304/f_{CPU}+1041	304/f_{CPU}+1813	禁止
设置信息功能		72/f_{CPU}+753218	72/f_{CPU}+753218	72/f_{CPU}+753990	使能
模式检查功能		30/f_{CPU}+952	30/f_{CPU}+952	30/f_{CPU}+1724	禁止
EEPROM 写入功能		268/f_{CPU}+1297+ 2286×W	268/f_{CPU}+1297+ 2286×W	268/f_{CPU}+2069+ 2286×W	使能

注 这是当自编程启动功能执行后立即执行该功能的功能处理时间。自编程启动功能之外的功能执行之后的处理时间，和 RSTOP = 0 的情况一样。

备注 RSTOP： 内部振荡模式寄存器（RCM）的第 0 位
RSTS： RCM 的第 7 位
MCS： 主时钟模式寄存器（MCM）的第 1 位
f_{CPU}： CPU 时钟频率
W： 将要写入的字的数量（1 字 = 4 字节）

表 **27-13.** 处理时间和中断响应（**4/4**）
（当静态模型库和入口 **RAM** 被分配在短直接寻址范围之内）

功能名称		处理时间（单位：μs）			中断响应
		RSTOP = 0 且 RSTS = 1 （在内部高速振荡器稳定操作期间）		RSTOP = 1 （内部高速振荡器停止）[注]	
		MCS = 0 （CPU 运行于内部高速振荡 时钟）	MCS = 1 （CPU 运行于高速系统时钟）	MCS = 1 （CPU 运行于高速系统时钟）	
自编程启动功能		$34/f_{CPU}$	$34/f_{CPU}$	$34/f_{CPU}$	禁止
自编程结束功能		$34/f_{CPU}$	$34/f_{CPU}$	$34/f_{CPU}$	禁止
初始化功能		$55/f_{CPU}+462$	$55/f_{CPU}+462$	$55/f_{CPU}+473$	禁止
块擦除功能		$136/f_{CPU}+352516$	$136/f_{CPU}+352516$	$136/f_{CPU}+352528$	使能
字写入功能		$272/f_{CPU}+477+2142\times W$	$272/f_{CPU}+477+2142\times W$	$272/f_{CPU}+488+2142\times W$	使能
块校验功能		$136/f_{CPU}+24918$	$136/f_{CPU}+24918$	$136/f_{CPU}+24930$	使能
块空白检测功能		$136/f_{CPU}+12128$	$136/f_{CPU}+12128$	$136/f_{CPU}+12139$	使能
获取信息功能	选项值 03H	$134/f_{CPU}+388$	$134/f_{CPU}+388$	$134/f_{CPU}+399$	禁止
	选项值 04H	$144/f_{CPU}+378$	$144/f_{CPU}+378$	$144/f_{CPU}+390$	禁止
	选项值 05H	$304/f_{CPU}+363$	$304/f_{CPU}+363$	$304/f_{CPU}+375$	禁止
设置信息功能		$72/f_{CPU}+752540$	$72/f_{CPU}+752540$	$72/f_{CPU}+753654$	使能
模式检查功能		$30/f_{CPU}+274$	$30/f_{CPU}+274$	$30/f_{CPU}+286$	禁止
EEPROM 写入功能		$268/f_{CPU}+619+2286\times W$	$268/f_{CPU}+619+2286\times W$	$268/f_{CPU}+630+2286\times W$	使能

注　这是当自编程启动功能执行后立即执行该功能的功能处理时间。自编程启动功能之外的功能执行之后的处理时间，和 RSTOP = 0 的情况一样。

备注　RSTOP：内部振荡模式寄存器（RCM）的第 0 位
RSTS：RCM 的第 7 位
MCS：主时钟模式寄存器（MCM）的第 1 位
f_{CPU}：CPU 时钟频率
W：将要写入的字的数量（1 字 = 4 字节）

表 27-14. 中断响应时间（正常模式的库）（1/2）

功能名称	中断响应时间（单位：μs）					
	当入口 RAM 被分配在短直接寻址范围之外时			当入口 RAM 被分配在短直接寻址范围之内时		
	RSTOP = 0 和 RSTS = 1（在内部高速振荡器稳定操作期间）		RSTOP = 1（内部高速振荡器停止）[注]	RSTOP = 0 和 RSTS = 1（在内部高速振荡器稳定操作期间）		RSTOP = 1（内部高速振荡器停止）[注]
	MCS = 0（CPU 运行于内部高速振荡时钟）	MCS = 1（CPU 运行于高速系统时钟）	MCS = 1（CPU 运行于高速系统时钟）	MCS = 0（CPU 运行于内部高速振荡时钟）	MCS = 1（CPU 运行于高速系统时钟）	MCS = 1（CPU 运行于高速系统时钟）
块擦除功能	179/f_{CPU}+1269	179/f_{CPU}+1269	179/f_{CPU}+1912	179/f_{CPU}+703	179/f_{CPU}+703	179/f_{CPU}+713
字写入功能	333/f_{CPU}+1098	333/f_{CPU}+1098	333/f_{CPU}+1742	333/f_{CPU}+533	333/f_{CPU}+533	333/f_{CPU}+543
块校验功能	179/f_{CPU}+1013	179/f_{CPU}+1013	179/f_{CPU}+1656	179/f_{CPU}+448	179/f_{CPU}+448	179/f_{CPU}+456
块空白检测功能	179/f_{CPU}+993	179/f_{CPU}+993	179/f_{CPU}+1637	179/f_{CPU}+428	179/f_{CPU}+428	179/f_{CPU}+438
设置信息功能	80/f_{CPU}+833	80/f_{CPU}+833	80/f_{CPU}+1477	80/f_{CPU}+346	80/f_{CPU}+346	80/f_{CPU}+346
EEPROM 写入功能	333/f_{CPU}+1107	333/f_{CPU}+1107	333/f_{CPU}+1751	333/f_{CPU}+542	333/f_{CPU}+542	333/f_{CPU}+552

注　这是当自编程启动功能执行后立即执行该功能的功能处理时间。自编程启动功能之外的功能执行之后的处理时间，和 RSTOP = 0 的情况一样。

备注　RSTOP：内部振荡模式寄存器（RCM）的第 0 位
RSTS：RCM 的第 7 位
MCS：主时钟模式寄存器（MCM）的第 1 位
f_{CPU}：CPU 时钟频率
W：将要写入的字的数量（1 字 = 4 字节）

表 27-14. 中断响应时间（静态模式的库）（2/2）

功能名称	中断响应时间（μs）					
	当入口 RAM 被分配在短直接寻址范围之外时			当入口 RAM 被分配在短直接寻址范围之内时		
	RSTOP = 0 和 RSTS = 1（在内部高速振荡器稳定操作期间）		RSTOP = 1（内部高速振荡器停止）[注]	RSTOP = 0 和 RSTS = 1（在内部高速振荡器稳定操作期间）		RSTOP = 1（内部高速振荡器停止）[注]
	MCS = 0（CPU 运行于内部高速振荡时钟）	MCS = 1（CPU 运行于高速系统时钟）	MCS = 1（CPU 运行于高速系统时钟）	MCS = 0（CPU 运行于内部高速振荡时钟）	MCS = 1（CPU 运行于高速系统时钟）	MCS = 1（CPU 运行于高速系统时钟）
块擦除功能	136/f_{CPU}+1269	136/f_{CPU}+1269	136/f_{CPU}+1912	136/f_{CPU}+703	136/f_{CPU}+703	136/f_{CPU}+713
字写入功能	272/f_{CPU}+1098	272/f_{CPU}+1098	272/f_{CPU}+1742	272/f_{CPU}+533	272/f_{CPU}+533	272/f_{CPU}+543
块校验功能	136/f_{CPU}+1013	136/f_{CPU}+1013	136/f_{CPU}+1656	136/f_{CPU}+448	136/f_{CPU}+448	136/f_{CPU}+456
块空白检测功能	136/f_{CPU}+993	136/f_{CPU}+993	136/f_{CPU}+1637	136/f_{CPU}+428	136/f_{CPU}+428	136/f_{CPU}+438
设置信息功能	72/f_{CPU}+833	72/f_{CPU}+833	72/f_{CPU}+1477	72/f_{CPU}+346	72/f_{CPU}+346	72/f_{CPU}+346
EEPROM 写入功能	268/f_{CPU}+1107	268/f_{CPU}+1107	268/f_{CPU}+1751	268/f_{CPU}+542	268/f_{CPU}+542	268/f_{CPU}+552

注　这是当自编程启动功能执行后立即执行该功能的功能处理时间。自编程启动功能之外的功能执行之后的处理时间，和 RSTOP = 0 的情况一样。

备注　RSTOP：内部振荡模式寄存器（RCM）的第 0 位
RSTS：RCM 的第 7 位
MCS：主时钟模式寄存器（MCM）的第 1 位
f_{CPU}：CPU 时钟频率
W：将要写入的字的数量（1 字 = 4 字节）

27.10.1 引导交换功能

在自编程期间，如果由于电源失效或其它原因导致引导区域的重写操作失败，则引导区域的数据可能会丢失，且不能通过复位重启程序。

使用引导交换功能可以避免这个问题。

在通过自编程擦除引导簇 0 [注]（这是一个引导程序区域）之前，提前将一个新的引导程序写入引导簇 1。当该程序正确写入引导簇 1 后，通过使用 78K0/LE3 固件的设置信息功能将引导簇 1 与引导簇 0 的内容交换，这样引导簇 1 被用作引导区域。之后，对原有的引导程序区域引导簇 0 进行擦除或写入。

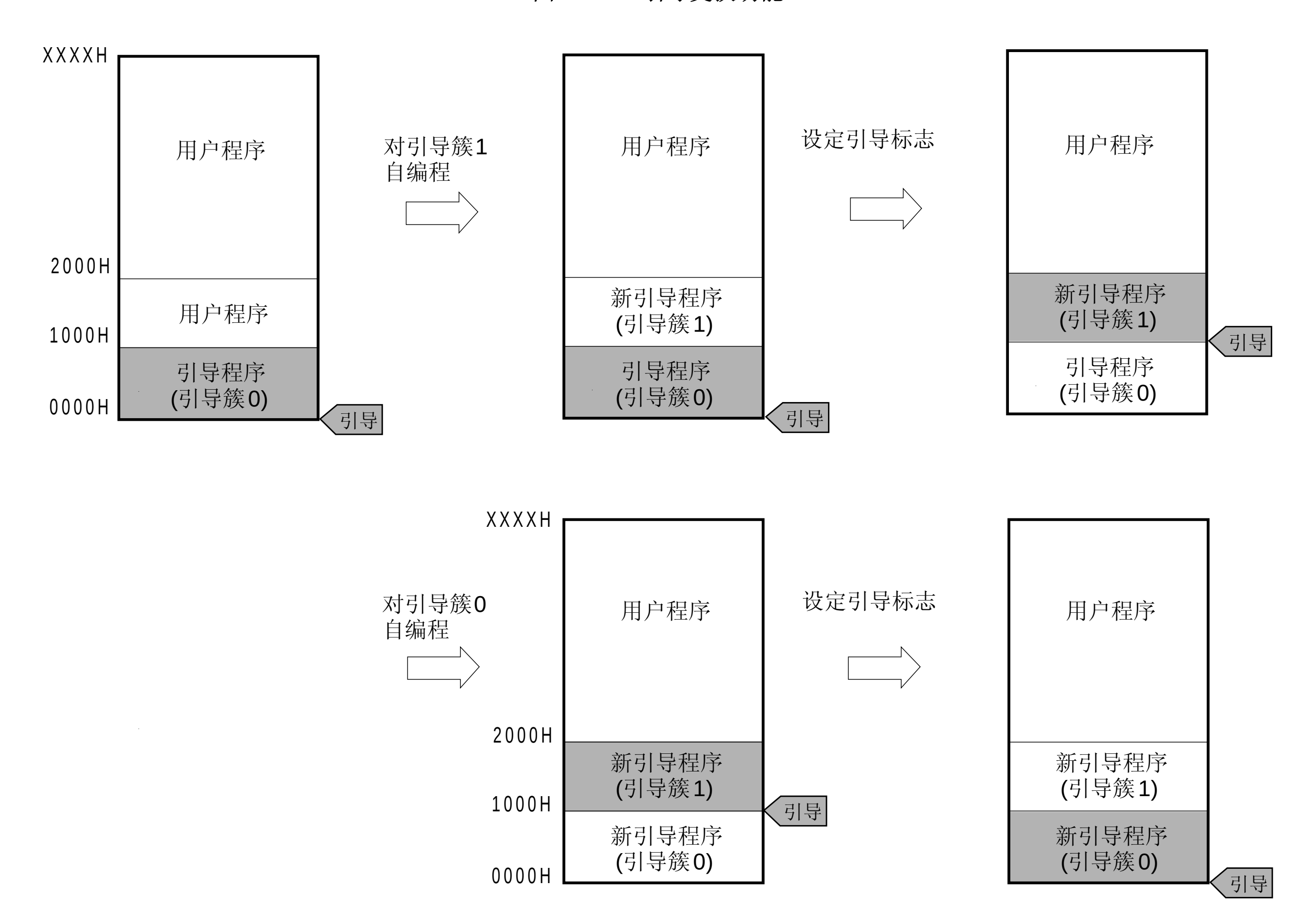
这样，即使在重写引导程序区域时出现电源失效，程序也能够正确执行，因为程序在复位并再次启动时，它是从已经交换的引导簇 1 的启动引导程序。

如果程序已经正确写入引导簇 0，则通过使用 78K0/LE3 固件的设置信息功能来恢复原有引导区域。

注　　一个引导簇的容量为 4 KB，通过引导交换功能交换引导簇 0 与引导簇 1。

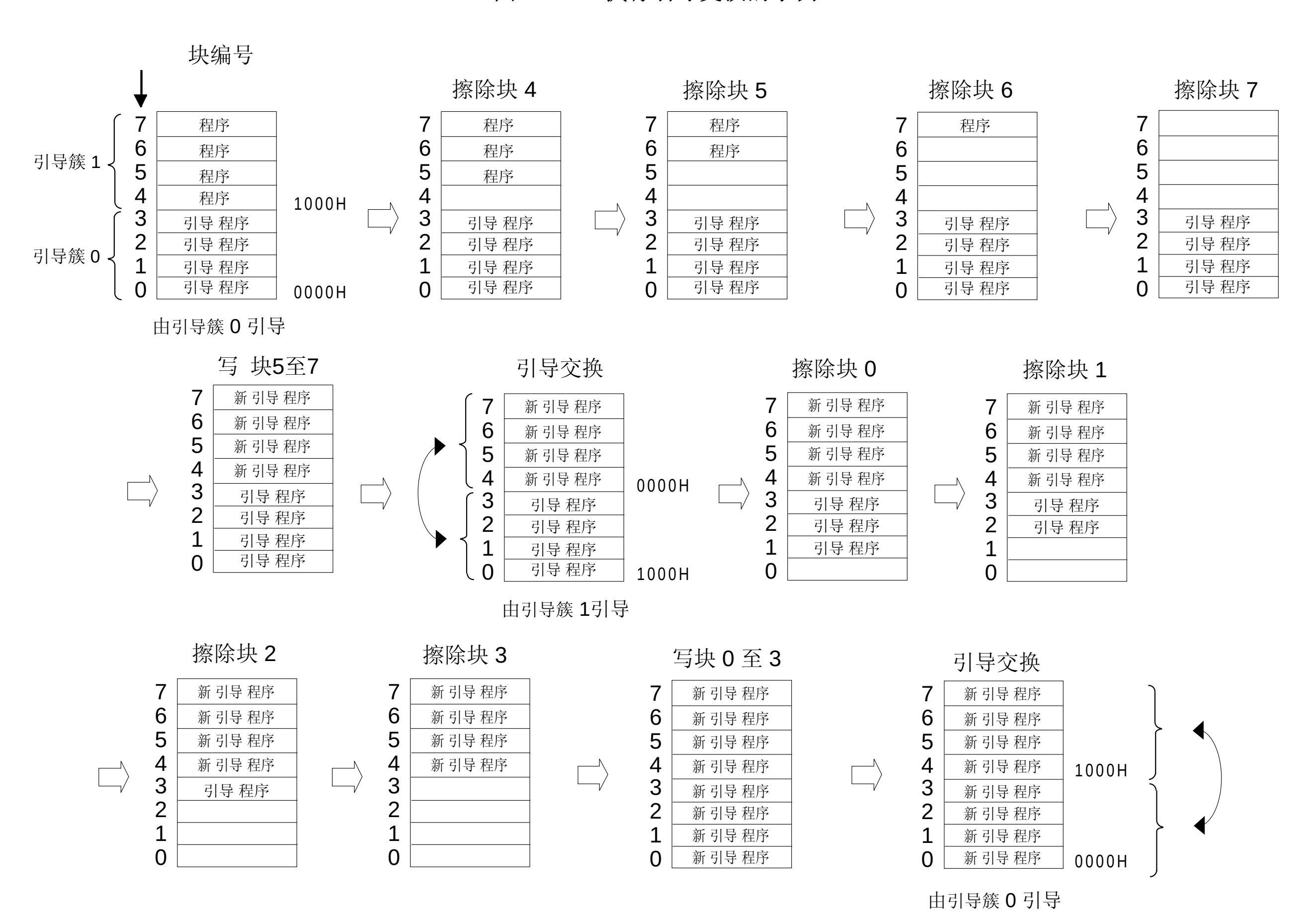
引导簇 0（0000H 至 0FFFH）：　原有的引导程序区域
引导簇 1（1000H 至 1FFFH）：　要进行引导交换的区域

图 27-17. 引导交换功能



备注　　设置引导标志后，当产生复位时，引导簇 1 变成 0000H 至 0FFFH。

图 27-18. 执行引导交换的示例

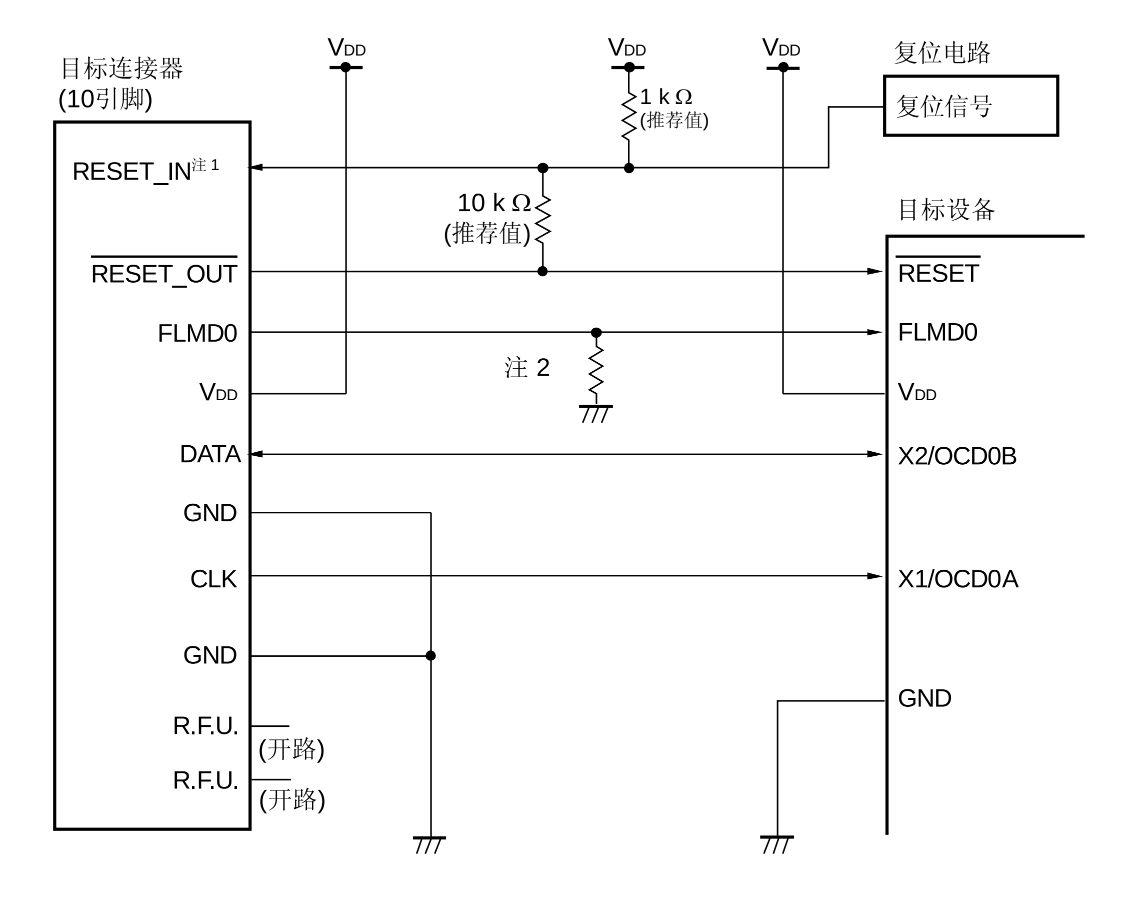


28.1 QB-MINI2 与 78K0/LE3 的连接

通过片上调试仿真器（QB-MINI2），78K0/LE3 使用 V_{DD}、FLMD0、$\overline{\text{RESET}}$、OCD0A/X1、OCD0B/X2 和 V_{SS} 引脚与主机通信。

注意事项 **78K0/LE3** 具有片上调试功能，该功能的提供是用于开发和评估。不要在设计用于大规模量产的产品上使用片上调试功能，因为使用该功能的话，可能会超过 **flash** 存储器可重写的保证次数，因此产品的可靠性无法保证。**NEC** 电子不对使用片上调试功能时发生的问题负责。

图 **28-1. QB-MINI2** 与 **78K0/LE3** 的连接示例



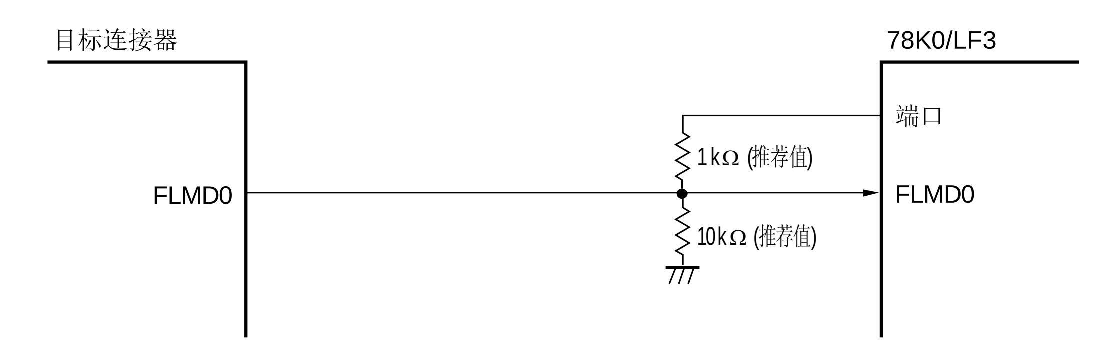
注　**1.** 该连接的设计是假设复位信号从 N-ch 开漏缓冲器输出。（输出电阻：100 Ω）详情参见 **QB-MINI2** 用户手册（**U18371E**）。

2. 下拉电阻值至少应为 470 Ω （推荐使用 10 kΩ）。

注意事项　在片上调试期间，从 **OCD0A/X1** 引脚输入时钟。

当通过片上调试执行自编程时，按下列方式连接 FLMD0 引脚。

图 28-2. 通过片上调试的方法执行自编程时 FLMD0 引脚的连接



28.2 QB-MINI2 使用的保留区域

QB-MINI2 使用如图 28-3 所示的保留区域来实现和 78K0/LE3 的通信，或各种调试功能。阴影保留区域是用于各种调试功能，其他保留区域是调试总是需要的区域。这些保留区域可以使用用户程序和编译器选项来预留。

在自编程期间使用引导交换功能时，预先设置相同的值到引导簇 1。

保留区域的详细信息，参见 **QB-MINI2** 用户手册（**U18371E**）。

图 28-3. QB-MINI2 使用的保留区域

内部 ROM 空间

伪 RRM 区域
(256 字节)

调试监控区域
(257 字节)

安全 ID 区域
(10 字节)

选项字节区域 (1 字节)

软件中断区域 (2 字节)

调试监控区域 (2 字节)

内部 RAM 空间

调试所需的堆栈区域
(最大值 16 字节)

FF7FH

伪 RRM 区域
(16 字节)注

F7F0H

备注　阴影保留区域：　各个调试功能需要使用的区域

　　　　其他保留区域：　调试总是需要的区域

第二十九章　指令集

本章以表格方式列出了 78K0/LE3 的指令集。每种操作和操作代码的信息，参见另外的文件 **78K/0 系列指令用户手册(U12326E)**。

29.1 操作列表的约定

29.1.1 操作数标识符和说明方法

按照指令操作数标识符的说明方法（详情参见汇编器规范），在每种指令的"操作数"栏写入操作数。如果有两种或更多的方法，可选其中之一。大写字母和符号#、!、$以及 [ ]是关键字，必须按其原样书写。每种符号的含义如下。

- #：　立即数标识
- !：　绝对地址标识
- $：　相对地址标识
- [ ]：　间接地址标识

立即数的情况，用来描述一个适当的数值型数据或标号。当使用标号时，注意必须写入符号#、!、$以及 [ ]符号。

对于操作数寄存器标识符 r 和 rp，功能名称 （X，A，C 等）或绝对名称（下表括号中的名称： R0，R1，R2 等）都可用于标识说明。

表 29-1. 操作数标识符和说明方法

标识符	说明方法
r rp sfr sfrp	X (R0), A (R1), C (R2), B (R3), E (R4), D (R5), L (R6), H (R7) AX (RP0), BC (RP1), DE (RP2), HL (RP3) 特殊功能寄存器符号[注] 特殊功能寄存器符号 （仅限 16 位可操作寄存器的偶地址）[注]
saddr saddrp	FE20H 至 FF1FH 立即数或标号 FE20H 至 FF1FH 立即数或标号 （仅限偶地址）
addr16 addr11 addr5	0000H 至 FFFFH 立即数或标号 （仅限 16 位数据传送指令的偶地址） 0800H 至 0FFFH 立即数或标号 0040H 至 007FH 立即数或标号 （仅限偶地址）
word byte bit	16 位立即数或标号 8 位立即数或标号 3 位立即数或标号
RBn	RB0 至 RB3

注　不能通过这些操作数访问地址 FFD0H 至 FFDFH。

备注　特殊功能寄存器符号参见表**表 3-6 特殊功能寄存器列表**。

29.1.2 操作栏描述

A：　A 寄存器；8 位累加器
X：　X 寄存器
B：　B 寄存器
C：　C 寄存器
D：　D 寄存器
E：　E 寄存器
H：　H 寄存器
L：　L 寄存器
AX：　AX 寄存器对；16 位累加器
BC：　BC 寄存器对
DE：　DE 寄存器对
HL：　HL 寄存器对
PC：　程序计数器
SP：　堆栈指针
PSW：　程序状态字
CY：　进位标志
AC：　辅助进位标志
Z：　零标志
RBS：　寄存器组选择标志
IE：　中断请求使能标志
( )：　括号中的地址或寄存器指示的存储单元的内容
X_H, X_L：　16 位寄存器的高 8 位和低 8 位
∧：　逻辑与 (AND)
∨：　逻辑或 (OR)
$\veebar$：　逻辑异或 (异 OR)
$\overline{}$：　数据取反
addr16：　16 位立即数或标号
jdisp8：　带符号的 8 位数据（偏移量）

29.1.3 标志操作栏的描述

(空)：　不受影响
0：　清除为 0
1：　设置为 1
×：　根据结果进行置位/清除
R：　恢复先前保存的值

29.2 操作列表

指令组	助记符	操作数	字节数	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位数据传送	MOV	r, #byte	2	4	–	r ← byte			
		saddr, #byte	3	6	7	(saddr) ← byte			
		sfr, #byte	3	–	7	sfr ← byte			
		A, r [注 3]	1	2	–	A ← r			
		r, A [注 3]	1	2	–	r ← A			
		A, saddr	2	4	5	A ← (saddr)			
		saddr, A	2	4	5	(saddr) ← A			
		A, sfr	2	–	5	A ← sfr			
		sfr, A	2	–	5	sfr ← A			
		A, !addr16	3	8	9	A ← (addr16)			
		!addr16, A	3	8	9	(addr16) ← A			
		PSW, #byte	3	–	7	PSW ← byte	×	×	×
		A, PSW	2	–	5	A ← PSW			
		PSW, A	2	–	5	PSW ← A	×	×	×
		A, [DE]	1	4	5	A ← (DE)			
		[DE], A	1	4	5	(DE) ← A			
		A, [HL]	1	4	5	A ← (HL)			
		[HL], A	1	4	5	(HL) ← A			
		A, [HL + byte]	2	8	9	A ← (HL + byte)			
		[HL + byte], A	2	8	9	(HL + byte) ← A			
		A, [HL + B]	1	6	7	A ← (HL + B)			
		[HL + B], A	1	6	7	(HL + B) ← A			
		A, [HL + C]	1	6	7	A ← (HL + C)			
		[HL + C], A	1	6	7	(HL + C) ← A			
	XCH	A, r [注 3]	1	2	–	A ↔ r			
		A, saddr	2	4	6	A ↔ (saddr)			
		A, sfr	2	–	6	A ↔ (sfr)			
		A, !addr16	3	8	10	A ↔ (addr16)			
		A, [DE]	1	4	6	A ↔ (DE)			
		A, [HL]	1	4	6	A ↔ (HL)			
		A, [HL + byte]	2	8	10	A ↔ (HL + byte)			
		A, [HL + B]	2	8	10	A ↔ (HL + B)			
		A, [HL + C]	2	8	10	A ↔ (HL + C)			

注 **1.** 当访问内部高速 RAM 时，或针对无数据访问的指令

2. 当访问内部高速 RAM 以外的区域时

3. “r = A”除外

备注 **1.** 一个指令时钟周期是由处理器时钟控制寄存器（PCC）选择的 CPU 时钟（f_{CPU}）的一个周期。

2. 该时钟周期适适用于内部 ROM 程序。

指令组	助记符	操作数	字节数	时钟		操作	标志
				注 1	注 2		Z AC CY
16 位数据传送	**MOVW**	rp, #word	3	6	–	rp ← word	
		saddrp, #word	4	8	10	(saddrp) ← word	
		sfrp, #word	4	–	10	sfrp ← word	
		AX, saddrp	2	6	8	AX ← (saddrp)	
		saddrp, AX	2	6	8	(saddrp) ← AX	
		AX, sfrp	2	–	8	AX ← sfrp	
		sfrp, AX	2	–	8	sfrp ← AX	
		AX, rp 注 3	1	4	–	AX ← rp	
		rp, AX 注 3	1	4	–	rp ← AX	
		AX, !addr16	3	10	12	AX ← (addr16)	
		!addr16, AX	3	10	12	(addr16) ← AX	
	XCHW	AX, rp 注 3	1	4	–	AX ↔ rp	
8 位操作	**ADD**	A, #byte	2	4	–	A, CY ← A + byte	× × ×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) + byte	× × ×
		A, r 注 4	2	4	–	A, CY ← A + r	× × ×
		r, A	2	4	–	r, CY ← r + A	× × ×
		A, saddr	2	4	5	A, CY ← A + (saddr)	× × ×
		A, !addr16	3	8	9	A, CY ← A + (addr16)	× × ×
		A, [HL]	1	4	5	A, CY ← A + (HL)	× × ×
		A, [HL + byte]	2	8	9	A, CY ← A + (HL + byte)	× × ×
		A, [HL + B]	2	8	9	A, CY ← A + (HL + B)	× × ×
		A, [HL + C]	2	8	9	A, CY ← A + (HL + C)	× × ×
	ADDC	A, #byte	2	4	–	A, CY ← A + byte + CY	× × ×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) + byte + CY	× × ×
		A, r 注 4	2	4	–	A, CY ← A + r + CY	× × ×
		r, A	2	4	–	r, CY ← r + A + CY	× × ×
		A, saddr	2	4	5	A, CY ← A + (saddr) + CY	× × ×
		A, !addr16	3	8	9	A, CY ← A + (addr16) + C	× × ×
		A, [HL]	1	4	5	A, CY ← A + (HL) + CY	× × ×
		A, [HL + byte]	2	8	9	A, CY ← A + (HL + byte) + CY	× × ×
		A, [HL + B]	2	8	9	A, CY ← A + (HL + B) + CY	× × ×
		A, [HL + C]	2	8	9	A, CY ← A + (HL + C) + CY	× × ×

注　**1.** 当访问内部高速 RAM 时，或针对无数据访问的指令
2. 当访问内部高速 RAM 以外的区域时
3. 仅当 rp = BC，DE 或 HL
4. “r = A”除外

备注　**1.** 一个指令时钟周期是由处理器时钟控制寄存器（PCC）选择的 CPU 时钟（f_{CPU}）的一个周期。
2. 该时钟周期适用于内部 ROM 程序。

指令组	助记符	操作数	字节数	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位操作	**SUB**	A, #byte	2	4	–	A, CY ← A – byte	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) – byte	×	×	×
		A, r 注 3	2	4	–	A, CY ← A – r	×	×	×
		r, A	2	4	–	r, CY ← r – A	×	×	×
		A, saddr	2	4	5	A, CY ← A – (saddr)	×	×	×
		A, !addr16	3	8	9	A, CY ← A – (addr16)	×	×	×
		A, [HL]	1	4	5	A, CY ← A – (HL)	×	×	×
		A, [HL + byte]	2	8	9	A, CY ← A – (HL + byte)	×	×	×
		A, [HL + B]	2	8	9	A, CY ← A – (HL + B)	×	×	×
		A, [HL + C]	2	8	9	A, CY ← A – (HL + C)	×	×	×
	SUBC	A, #byte	2	4	–	A, CY ← A – byte – CY	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) – byte – CY	×	×	×
		A, r 注 3	2	4	–	A, CY ← A – r – CY	×	×	×
		r, A	2	4	–	r, CY ← r – A – CY	×	×	×
		A, saddr	2	4	5	A, CY ← A – (saddr) – CY	×	×	×
		A, !addr16	3	8	9	A, CY ← A – (addr16) – CY	×	×	×
		A, [HL]	1	4	5	A, CY ← A – (HL) – CY	×	×	×
		A, [HL + byte]	2	8	9	A, CY ← A – (HL + byte) – CY	×	×	×
		A, [HL + B]	2	8	9	A, CY ← A – (HL + B) – CY	×	×	×
		A, [HL + C]	2	8	9	A, CY ← A – (HL + C) – CY	×	×	×
	AND	A, #byte	2	4	–	A ← A ∧ byte	×		
		saddr, #byte	3	6	8	(saddr) ← (saddr) ∧ byte	×		
		A, r 注 3	2	4	–	A ← A ∧ r	×		
		r, A	2	4	–	r ← r ∧ A	×		
		A, saddr	2	4	5	A ← A ∧ (saddr)	×		
		A, !addr16	3	8	9	A ← A ∧ (addr16)	×		
		A, [HL]	1	4	5	A ← A ∧ (HL)	×		
		A, [HL + byte]	2	8	9	A ← A ∧ (HL + byte)	×		
		A, [HL + B]	2	8	9	A ← A ∧ (HL + B)	×		
		A, [HL + C]	2	8	9	A ← A ∧ (HL + C)	×		

注　**1.** 当访问内部高速 RAM 时，或针对无数据访问的指令
　　2. 当访问内部高速 RAM 以外的区域时
　　3. “r = A”除外

备注　**1.** 一个指令时钟周期是由处理器时钟控制寄存器（PCC）选择的 CPU 时钟（f_{CPU}）的一个周期。
　　　2. 该时钟周期适用于内部 ROM 程序。

指令组	助记符	操作数	字节数	时钟		操作	标志		
				注 1	**注 2**		Z	AC	CY
8 位操作	**OR**	A, #byte	2	4	–	A ← A ∨ byte	×		
		saddr, #byte	3	6	8	(saddr) ← (saddr) ∨ byte	×		
		A, r [注 3]	2	4	–	A ← A ∨ r	×		
		r, A	2	4	–	r ← r ∨ A	×		
		A, saddr	2	4	5	A ← A ∨ (saddr)	×		
		A, !addr16	3	8	9	A ← A ∨ (addr16)	×		
		A, [HL]	1	4	5	A ← A ∨ (HL)	×		
		A, [HL + byte]	2	8	9	A ← A ∨ (HL + byte)	×		
		A, [HL + B]	2	8	9	A ← A ∨ (HL + B)	×		
		A, [HL + C]	2	8	9	A ← A ∨ (HL + C)	×		
	XOR	A, #byte	2	4	–	A ← A ⊻ byte	×		
		saddr, #byte	3	6	8	(saddr) ← (saddr) ⊻ byte	×		
		A, r [注 3]	2	4	–	A ← A ⊻ r	×		
		r, A	2	4	–	r ← r ⊻ A	×		
		A, saddr	2	4	5	A ← A ⊻ (saddr)	×		
		A, !addr16	3	8	9	A ← A ⊻ (addr16)	×		
		A, [HL]	1	4	5	A ← A ⊻ (HL)	×		
		A, [HL + byte]	2	8	9	A ← A ⊻ (HL + byte)	×		
		A, [HL + B]	2	8	9	A ← A ⊻ (HL + B)	×		
		A, [HL + C]	2	8	9	A ← A ⊻ (HL + C)	×		
	CMP	A, #byte	2	4	–	A – byte	×	×	×
		saddr, #byte	3	6	8	(saddr) – byte	×	×	×
		A, r [注 3]	2	4	–	A – r	×	×	×
		r, A	2	4	–	r – A	×	×	×
		A, saddr	2	4	5	A – (saddr)	×	×	×
		A, !addr16	3	8	9	A – (addr16)	×	×	×
		A, [HL]	1	4	5	A – (HL)	×	×	×
		A, [HL + byte]	2	8	9	A – (HL + byte)	×	×	×
		A, [HL + B]	2	8	9	A – (HL + B)	×	×	×
		A, [HL + C]	2	8	9	A – (HL + C)	×	×	×

注　**1.** 当访问内部高速 RAM 时，或针对无数据访问的指令
　　2. 当访问内部高速 RAM 以外的区域时
　　3. “r = A”除外

备注　**1.** 一个指令时钟周期是由处理器时钟控制寄存器（PCC）选择的 CPU 时钟（f_{CPU}）的一个周期。
　　　2. 该时钟周期适用于内部 ROM 程序。

指令组	助记符	操作数	字节数	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
16 位操作	**ADDW**	AX, #word	3	6	–	AX, CY ← AX + word	×	×	×
	SUBW	AX, #word	3	6	–	AX, CY ← AX – word	×	×	×
	CMPW	AX, #word	3	6	–	AX – word	×	×	×
乘法/除法	**MULU**	X	2	16	–	AX ← A × X			
	DIVUW	C	2	25	–	AX (商), C (余数) ← AX ÷ C			
递增/递减	**INC**	r	1	2	–	r ← r + 1	×	×	
		saddr	2	4	6	(saddr) ← (saddr) + 1	×	×	
	DEC	r	1	2	–	r ← r – 1	×	×	
		saddr	2	4	6	(saddr) ← (saddr) – 1	×	×	
	INCW	rp	1	4	–	rp ← rp + 1			
	DECW	rp	1	4	–	rp ← rp – 1			
循环	**ROR**	A, 1	1	2	–	(CY, A_7 ← A_0, A_{m-1} ← A_m) × 1 次			×
	ROL	A, 1	1	2	–	(CY, A_0 ← A_7, A_{m+1} ← A_m) × 1 次			×
	RORC	A, 1	1	2	–	(CY ← A_0, A_7 ← CY, A_{m-1} ← A_m) × 1 次			×
	ROLC	A, 1	1	2	–	(CY ← A_7, A_0 ← CY, A_{m+1} ← A_m) × 1 次			×
	ROR4	[HL]	2	10	12	A_{3-0} ← $(HL)_{3-0}$, $(HL)_{7-4}$ ← A_{3-0}, $(HL)_{3-0}$ ← $(HL)_{7-4}$			
	ROL4	[HL]	2	10	12	A_{3-0} ← $(HL)_{7-4}$, $(HL)_{3-0}$ ← A_{3-0}, $(HL)_{7-4}$ ← $(HL)_{3-0}$			
BCD 调整	**ADJBA**		2	4	–	加法后的十进制调整	×	×	×
	ADJBS		2	4	–	减法后的十进制调整	×	×	×
位操作	**MOV1**	CY, saddr.bit	3	6	7	CY ← (saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← sfr.bit			×
		CY, A.bit	2	4	–	CY ← A.bit			×
		CY, PSW.bit	3	–	7	CY ← PSW.bit			×
		CY, [HL].bit	2	6	7	CY ← (HL).bit			×
		saddr.bit, CY	3	6	8	(saddr.bit) ← CY			
		sfr.bit, CY	3	–	8	sfr.bit ← CY			
		A.bit, CY	2	4	–	A.bit ← CY			
		PSW.bit, CY	3	–	8	PSW.bit ← CY	×	×	
		[HL].bit, CY	2	6	8	(HL).bit ← CY			

注 **1.** 当访问内部高速 RAM 时，或针对无数据访问的指令
2. 当访问内部高速 RAM 以外的区域时

备注 **1.** 一个指令时钟周期是由处理器时钟控制寄存器（PCC）选择的 CPU 时钟（f_{CPU}）的一个周期。
2. 该时钟周期适用于内部 ROM 程序。

指令组	助记符	操作	字节数	时钟		操作	标志		
				注 1	**注 2**		Z	AC	CY
位操作	**AND1**	CY, saddr.bit	3	6	7	CY ← CY ∧ (saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← CY ∧ sfr.bit			×
		CY, A.bit	2	4	–	CY ← CY ∧ A.bit			×
		CY, PSW.bit	3	–	7	CY ← CY ∧ PSW.bit			×
		CY, [HL].bit	2	6	7	CY ← CY ∧ (HL).bit			×
	OR1	CY, saddr.bit	3	6	7	CY ← CY ∨ (saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← CY ∨ sfr.bit			×
		CY, A.bit	2	4	–	CY ← CY ∨ A.bit			×
		CY, PSW.bit	3	–	7	CY ← CY ∨ PSW.bit			×
		CY, [HL].bit	2	6	7	CY ← CY ∨ (HL).bit			×
	XOR1	CY, saddr.bit	3	6	7	CY ← CY ⊻ (saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← CY ⊻ sfr.bit			×
		CY, A.bit	2	4	–	CY ← CY ⊻ A.bit			×
		CY, PSW. bit	3	–	7	CY ← CY ⊻ PSW.bit			×
		CY, [HL].bit	2	6	7	CY ← CY ⊻ (HL).bit			×
	SET1	saddr.bit	2	4	6	(saddr.bit) ← 1			
		sfr.bit	3	–	8	sfr.bit ← 1			
		A.bit	2	4	–	A.bit ← 1			
		PSW.bit	2	–	6	PSW.bit ← 1	×	×	×
		[HL].bit	2	6	8	(HL).bit ← 1			
	CLR1	saddr.bit	2	4	6	(saddr.bit) ← 0			
		sfr.bit	3	–	8	sfr.bit ← 0			
		A.bit	2	4	–	A.bit ← 0			
		PSW.bit	2	–	6	PSW.bit ← 0	×	×	×
		[HL].bit	2	6	8	(HL).bit ← 0			
	SET1	CY	1	2	–	CY ← 1			1
	CLR1	CY	1	2	–	CY ← 0			0
	NOT1	CY	1	2	–	CY ← CY			×

注　**1.**　当访问内部高速 RAM 时，或针对无数据访问的指令
　　2.　当访问内部高速 RAM 以外的区域时

备注　**1.**　一个指令时钟周期是由处理器时钟控制寄存器（PCC）选择的 CPU 时钟（f_{CPU}）的一个周期。
　　2.　该时钟周期适用于内部 ROM 程序。

指令组	助记符	操作数	字节数	时钟		操作	标志
				注 1	注 2		Z AC CY
调用/返回	**CALL**	!addr16	3	7	–	(SP – 1) ← $(PC + 3)_H$, (SP – 2) ← $(PC + 3)_L$, PC ← addr16, SP ← SP – 2	
	CALLF	!addr11	2	5	–	(SP – 1) ← $(PC + 2)_H$, (SP – 2) ← $(PC + 2)_L$, PC_{15-11} ← 00001, PC_{10-0} ← addr11, SP ← SP – 2	
	CALLT	[addr5]	1	6	–	(SP – 1) ← $(PC + 1)_H$, (SP – 2) ← $(PC + 1)_L$, PC_H ← (00000000, addr5 + 1), PC_L ← (00000000, addr5), SP ← SP – 2	
	BRK		1	6	–	(SP – 1) ← PSW, (SP – 2) ← $(PC + 1)_H$, (SP – 3) ← $(PC + 1)_L$, PC_H ← (003FH), PC_L ← (003EH), SP ← SP – 3, IE ← 0	
	RET		1	6	–	PC_H ← (SP + 1), PC_L ← (SP), SP ← SP + 2	
	RETI		1	6	–	PC_H ← (SP + 1), PC_L ← (SP), PSW ← (SP + 2), SP ← SP + 3	R R R
	RETB		1	6	–	PC_H ← (SP + 1), PC_L ← (SP), PSW ← (SP + 2), SP ← SP + 3	R R R
堆栈操作	**PUSH**	PSW	1	2	–	(SP – 1) ← PSW, SP ← SP – 1	
		rp	1	4	–	(SP – 1) ← rp_H, (SP – 2) ← rp_L, SP ← SP – 2	
	POP	PSW	1	2	–	PSW ← (SP), SP ← SP + 1	R R R
		rp	1	4	–	rp_H ← (SP + 1), rp_L ← (SP), SP ← SP + 2	
	MOVW	SP, #word	4	–	10	SP ← word	
		SP, AX	2	–	8	SP ← AX	
		AX, SP	2	–	8	AX ← SP	
无条件转移	**BR**	!addr16	3	6	–	PC ← addr16	
		$addr16	2	6	–	PC ← PC + 2 + jdisp8	
		AX	2	8	–	PCH ← A, PC_L ← X	
条件转移	**BC**	$addr16	2	6	–	如果 CY = 1，则 PC ← PC + 2 + jdisp8	
	BNC	$addr16	2	6	–	如果 CY = 0，则 PC ← PC + 2 + jdisp8	
	BZ	$addr16	2	6	–	如果 Z = 1，则 PC ← PC + 2 + jdisp8	
	BNZ	$addr16	2	6	–	如果 Z = 0，则 PC ← PC + 2 + jdisp8	

注　**1.** 当访问内部高速 RAM 时，或针对无数据访问的指令

　　2. 当访问内部高速 RAM 以外的区域时

备注　**1.** 一个指令时钟周期是由处理器时钟控制寄存器（PCC）选择的 CPU 时钟（f_{CPU}）的一个周期。

　　　2. 该时钟周期适用于内部 ROM 程序。

指令组	助记符	操作数	字节数	时钟		操作	标志
				注 1	注 2		Z AC CY
条件转移	**BT**	saddr.bit, $addr16	3	8	9	如果 (saddr.bit) = 1，则 PC ← PC + 3 + jdisp8	
		sfr.bit, $addr16	4	–	11	如果 sfr.bit = 1，则 PC ← PC + 4 + jdisp8	
		A.bit, $addr16	3	8	–	如果 A.bit = 1，则 PC ← PC + 3 + jdisp8	
		PSW.bit, $addr16	3	–	9	如果 PSW.bit = 1，则 PC ← PC + 3 + jdisp8	
		[HL].bit, $addr16	3	10	11	如果 (HL).bit = 1，则 PC ← PC + 3 + jdisp8	
	BF	saddr.bit, $addr16	4	10	11	如果 (saddr.bit) = 0，则 PC ← PC + 4 + jdisp8	
		sfr.bit, $addr16	4	–	11	如果 sfr.bit = 0，则 PC ← PC + 4 + jdisp8	
		A.bit, $addr16	3	8	–	如果 A.bit = 0，则 PC ← PC + 3 + jdisp8	
		PSW.bit, $addr16	4	–	11	如果 PSW. bit = 0，则 PC ← PC + 4 + jdisp8	
		[HL].bit, $addr16	3	10	11	如果 (HL).bit = 0，则 PC ← PC + 3 + jdisp8	
	BTCLR	saddr.bit, $addr16	4	10	12	PC ← PC + 4 + jdisp8，如果(saddr.bit) = 1，那么复位 (saddr.bit)	
		sfr.bit, $addr16	4	–	12	PC ← PC + 4 + jdisp8 if sfr.bit = 1 then reset sfr.bit	
		A.bit, $addr16	3	8	–	PC ← PC + 3 + jdisp8，如果 A.bit = 1，那么复位 A.bit	
		PSW.bit, $addr16	4	–	12	PC ← PC + 4 + jdisp8，如果 PSW.bit = 1，那么复位 PSW.bit	× × ×
		[HL].bit, $addr16	3	10	12	PC ← PC + 3 + jdisp8，如果 (HL).bit = 1，那么复位 (HL).bit	
	DBNZ	B, $addr16	2	6	–	B ← B – 1, 如果 B ≠ 0，那么 PC ← PC + 2 + jdisp8	
		C, $addr16	2	6	–	C ← C –1, 如果 C ≠ 0，那么 PC ← PC + 2 + jdisp8	
		saddr, $addr16	3	8	10	(saddr) ← (saddr) – 1,如果 (saddr) ≠ 0，那么 PC ← PC + 3 + jdisp8	
CPU 控制	**SEL**	RBn	2	4	–	RBS1, 0 ← n	
	NOP		1	2	–	无操作	
	EI		2	–	6	IE ← 1 (允许中断)	
	DI		2	–	6	IE ← 0 (禁止中断)	
	HALT		2	6	–	设置 HALT 模式	
	STOP		2	6	–	设置 STOP 模式	

注　**1.** 当访问内部高速 RAM 时，或针对无数据访问的指令
　　2. 当访问内部高速 RAM 以外的区域时

备注　**1.** 一个指令时钟周期是由处理器时钟控制寄存器（PCC）选择的 CPU 时钟（f_{CPU}）的一个周期。
　　　2. 该时钟周期适用于内部 ROM 程序。

29.3 按寻址类型列出指令

(1) 8 位指令

MOV, XCH, ADD, ADDC, SUB, SUBC, AND, OR, XOR, CMP, MULU, DIVUW, INC, DEC, ROR, ROL, RORC, ROLC, ROR4, ROL4, PUSH, POP, DBNZ

第 2 操作数 第 1 操作数	#byte	A	r 注	sfr	saddr	!addr16	PSW	[DE]	[HL]	[HL + byte] [HL + B] [HL + C]	$addr16	1	无
A	ADD ADDC SUB SUBC AND OR XOR CMP		MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV XCH	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV	MOV XCH	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP		ROR ROL RORC ROLC	
r	MOV	MOV ADD ADDC SUB SUBC AND OR XOR CMP											INC DEC
B, C											DBNZ		
sfr	MOV	MOV											
saddr	MOV ADD ADDC SUB SUBC AND OR XOR CMP	MOV									DBNZ		INC DEC
!addr16		MOV											
PSW	MOV	MOV											PUSH POP
[DE]		MOV											
[HL]		MOV											ROR4 ROL4
[HL + byte] [HL + B] [HL + C]		MOV											
X													MULU
C													DIVUW

注　　“r = A”除外

(2)　16 位指令

MOVW, XCHW, ADDW, SUBW, CMPW, PUSH, POP, INCW, DECW

第 2 操作数 / 第 1 操作数	#word	AX	rp [注]	sfrp	saddrp	!addr16	SP	无
AX	ADDW SUBW CMPW		MOVW XCHW	MOVW	MOVW	MOVW	MOVW	
rp	MOVW	MOVW [注]						INCW DECW PUSH POP
sfrp	MOVW	MOVW						
saddrp	MOVW	MOVW						
!addr16		MOVW						
SP	MOVW	MOVW						

注　　仅当 rp = BC, DE, HL 时

(3)　位操作指令

MOV1, AND1, OR1, XOR1, SET1, CLR1, NOT1, BT, BF, BTCLR

第 2 操作数 / 第 1 操作数	A.bit	sfr.bit	saddr.bit	PSW.bit	[HL].bit	CY	$addr16	无
A.bit						MOV1	BT BF BTCLR	SET1 CLR1
sfr.bit						MOV1	BT BF BTCLR	SET1 CLR1
saddr.bit						MOV1	BT BF BTCLR	SET1 CLR1
PSW.bit						MOV1	BT BF BTCLR	SET1 CLR1
[HL].bit						MOV1	BT BF BTCLR	SET1 CLR1
CY	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1			SET1 CLR1 NOT1

(4) 调用指令 / 跳转指令

CALL, CALLF, CALLT, BR, BC, BNC, BZ, BNZ, BT, BF, BTCLR, DBNZ

第 2 操作数 / 第 1 操作数	AX	!addr16	!addr11	[addr5]	$addr16
基本指令	BR	CALL BR	CALLF	CALLT	BR BC BNC BZ BNZ
复合指令					BT BF BTCLR DBNZ

(5) 其它指令

ADJBA, ADJBS, BRK, RET, RETI, RETB, SEL, NOP, EI, DI, HALT, STOP

<R> 注意事项 **78K0/LE3** 具有片上调试功能，该功能的提供是用于开发和评估。不要在量产时使用片上调试功能，因为使用该功能的话，可能会超过 **flash** 存储器可重写的保证次数，因此产品的可靠性无法保证。**NEC** 电子不对使用片上调试功能时发生的问题负责。

最大额定值（**T_A = 25°C**）（**1/2**）

参数	符号	条件	额定值	单位
供电电压	V_{DD}		–0.5 至+6.5	V
	V_{SS}		–0.5 至+0.3	V
	AV_{REF} 注 2		–0.5 至 V_{DD} + 0.3 注 1	V
	AV_{SS} 注 2		–0.5 至+0.3	V
REGC 引脚输入电压	V_{IREGC}		–0.5 至+ 3.6 和 –0.5 至 V_{DD}	V
输入电压	V_I	P11 至 P14，P20 至 P27，P31 至 P34，P40 至 P44，P80 至 P83，P100 至 P103, P110 至 P113, P120 至 P124, P140 至 P143，P150 至 P153，X1，X2，XT1，XT2，FLMD0，$\overline{RESET}$	–0.3 至 V_{DD} + 0.3 注 1	V
输出电压	V_O		–0.3 至 V_{DD} + 0.3 注 1	V
模拟输入电压	V_{AN}	ANI0 至 ANI7 注 2， DS0–至 DS2–注 3，DS0+至 DS2+注 3	–0.3 至 AV_{REF} + 0.3 注 1 和 –0.3 至 V_{DD} + 0.3 注 1	V
	REF+注 3		–0.5 至 AV_{REF} + 0.3 注 1	V
	REF–注 3		–0.5 至+ 0.3	V

注 **1.** 必须为 6.5 V 或更低。
2. 仅限μPD78F045x 和 78F046x。
3. 仅限μPD78F046x 。

注意事项 任何参数，即使瞬间超过最大额定值，都可能使产品性能受损。就是说，产品工作在最大额定值时处于物理损坏的临界点，所以，产品必须在不超出最大额定值的条件下使用。

备注 除非另外说明，复用功能引脚的特性与端口引脚的相同。

标准产品

最大额定值（T_A = 25°C）　（2/2）

参数	符号	条件		额定值	单位
输出电流，高	I_{OH1}	每一个引脚	P11 至 P14，P31 至 P34, P40 至 P44，P80 至 P83, P100 至 P103, P110 至 P113， P120, P140 至 P143, P150 至 P153	–10	mA
		所有引脚的总和 –35 mA	P11 至 P14，P31 至 P34, P40 至 P44，P120	–25	mA
			P80 至 P83, P100 至 P103, P110 至 P113, P140 至 P143, P150 至 P153	–10	mA
	I_{OH2}	每一个引脚	P20 至 P27	–0.5	mA
		所有引脚的总和		–2	mA
输出电流，低	I_{OL}	每一个引脚	P11 至 P14，P31 至 P34, P40 至 P44，P80 至 P83, P100 至 P103, P110 至 P113，P120, P140 至 P143，P150 至 P153	30	mA
		所有引脚的总和 80 mA	P11 至 P14，P31 至 P34, P40 至 P44，P120	40	mA
			P80 至 P83, P100 至 P103, P110 至 P113, P140 至 P143, P150 至 P153	40	mA
		每一个引脚	P20 至 P27	1	mA
		所有引脚的总和		5	mA
工作环境温度	T_A			–40 至+85	°C
存储温度	T_{stg}			–65 至+150	°C

注意事项　任何参数，即使瞬间超过最大额定值，都可能使产品性能受损。就是说，产品工作在最大额定值时处于物理损坏的临界点，所以，产品必须在不超出最大额定值的条件下使用。

备注　除非另外说明，复用功能引脚的特性与端口引脚的相同。

X1 振荡器特性

（T_A = –40 至+85°C，1.8 V ≤ V_{DD} ≤ 5.5 V，V_{SS} = AV_{SS} = 0 V）

振荡器	推荐电路	参数	条件	最小值	典型值	最大值	单位
	Vss X1 X2 C1 C2	X1 时钟振荡频率（fx）[注]	2.7 V ≤ V_{DD} ≤ 5.5 V	2.0		10.0	MHz
			1.8 V ≤ V_{DD} < 2.7 V	2.0		5.0	
晶体振荡器	Vss X1 X2 C1 C2	X1 时钟振荡频率（fx）[注]	2.7 V ≤ V_{DD} ≤ 5.5 V	2.0		10.0	MHz
			1.8 V ≤ V_{DD} < 2.7 V	2.0		5.0	

注　仅表示振荡器特性。指令执行时间参见 **AC** 特性。

注意事项 **1.** 在使用 **X1** 时钟时，上图中虚线框内的连线应遵从如下的连接方式，以避免线间电容的负面影响。

- 连线要保证尽可能短。
- 振荡器连线不要和其它的信号线互相交叉。
- 布线时不要将振荡器连线靠近通过大波动电流的信号线
- 保持振荡器电容的地电位和 **Vss** 相同。
- 不要将电容的接地点连接在大电流通过的地模式。
- 不要从振荡器取信号。

2. 复位释放后，**CPU** 由内部高速振荡时钟来启动，因此用户要使用振荡稳定时间计数器的状态寄存器（**OSTC**）检查 **X1** 时钟振荡稳定时间。在对所使用的振荡器的振荡稳定时间进行充分评估后，再由 **OSTC** 寄存器和振荡稳定时间选择寄存器（**OSTS**）决定振荡稳定时间。

标准产品

内部振荡特性

（T_A = –40 至+85°C，1.8 V ≤ V_{DD} ≤ 5.5 V，V_{SS} = AV_{SS} = 0 V）

振荡器	参数	条件		最小值	典型值	最大值	单位
8 MHz 内部振荡器	内部高速振荡时钟频率（f_{RH}）注 1，2	RSTS = 1	2.5 V ≤ V_{DD} ≤ 5.5 V	7.6	8.0	8.4	MHz
			1.8 V ≤ V_{DD} < 2.5 V	6.75	8.0	8.4	MHz
		RSTS = 0		2.48	5.6	9.86	MHz
240 kHz 内部振荡器	内部低速振荡时钟频率（f_{RL}）	2.6 V ≤ V_{DD} ≤ 5.5 V		216	240	264	kHz
		1.8 V ≤ V_{DD} < 2.6 V		192	240	264	kHz

注　**1.** 仅表示振荡器特性。指令执行时间参见 **AC** 特性。
　　2. 当设置 HIOTRM = 10H（±0%：默认）时

备注　RSTS：内部振荡模式寄存器（RCM）的第 7 位。

XT1 振荡器特性

（T_A = –40 至+85°C，1.8 V ≤ V_{DD} ≤ 5.5 V，V_{SS} = AV_{SS} = 0 V）

振荡器	推荐电路	参数	条件	最小值	典型值	最大值	单位
晶体振荡器	Vss XT2 XT1 / Rd / C4 C3	XT1 时钟振荡频率（f_{XT}）注		32	32.768	35	kHz

注　仅表示振荡器特性。指令执行时间参见 **AC** 特性。

注意事项　**1.**　在使用 **X1** 时钟时，上图中虚线框内的连线应遵从如下的连接方式，以避免线间电容的负面影响。

- 连线要保证尽可能短。
- 振荡器连线不要和其它的信号线互相交叉。
- 布线时不要将振荡器连线靠近通过大波动电流的信号线
- 保持振荡器电容的地电位和 **Vss** 相同。
- 不要将电容的接地点连接在大电流通过的地模式。
- 不要从振荡器取信号。

2.　**XT1** 振荡器设计作为一个低振幅电路，用于降低功耗，比 **X1** 振荡器更容易受到噪音干扰，因此在使用 **XT1** 时钟时应特别注意连线方式。

备注　关于谐振器的选择和振荡器常数，客户需要自己评估振荡性能，或者向振荡器制造商申请评估。

<R>　推荐振荡器常数

(1) X1 振荡器：陶瓷谐振器 (T_A = –40 至 +85°C)

制造商	部件编号	SMD/ Lead	频率 (MHz)	推荐电路常数		振荡电压范围	
				C1 (pF)	C2 (pF)	最小值 (V)	最大值 (V)
Murata Mfg.	CSTCC2M00G56-R0	SMD	2.00	内部 (47)	内部 (47)	1.8	5.5
	CSTLS4M00G56-B0	Lead	4.00	内部 (47)	内部 (47)		
	CSTCR4M00G55-R0	SMD		内部 (39)	内部 (39)		
	CSTLS4M19G56-B0	Lead	4.194	内部 (47)	内部 (47)		
	CSTCR4M19G55-R0	SMD		内部 (39)	内部 (39)		
	CSTLS4M91G56-B0	Lead	4.915	内部 (47)	内部 (47)	2.0	
	CSTCR4M91G55-R0	SMD		内部 (39)	内部 (39)	1.8	
	CSTLS5M00G56-B0	Lead	5.00	内部 (47)	内部 (47)	2.0	
	CSTCR5M00G55-R0	SMD		内部 (39)	内部 (39)	1.8	
	CSTLS6M00G56-B0	Lead	6.00	内部 (47)	内部 (47)	2.2	
	CSTCR6M00G55-R0	SMD		内部 (39)	内部 (39)	1.9	
	CSTLS8M00G56-B0	Lead	8.00	内部 (47)	内部 (47)	2.2	
	CSTCE8M00G55-R0	SMD		内部 (33)	内部 (33)	1.8	
	CSTLS8M38G56-B0	Lead	8.388	内部 (47)	内部 (47)	2.2	
	CSTCE8M38G55-R0	SMD		内部 (33)	内部 (33)	1.8	
	CSTLS10M0G53-B0	SMD	10.0	内部 (15)	内部 (15)	1.8	
	CSTCE10M0G55-R0	SMD		内部 (33)	内部 (33)	2.1	
Murata Mfg. (低电容产品)	CSTLS4M91G53-B0	Lead	4.915	内部 (15)	内部 (15)	1.8	5.5
	CSTLS5M00G53-B0	Lead	5.00	内部 (15)	内部 (15)	1.8	
	CSTCR6M00G53-R0	SMD	6.00	内部 (15)	内部 (15)	1.8	
	CSTLS6M00G53-B0	Lead		内部 (15)	内部 (15)	1.8	
	CSTLS8M00G53-B0	Lead	8.00	内部 (15)	内部 (15)	1.8	
	CSTLS8M38G53-B0	Lead	8.388	内部 (15)	内部 (15)	1.8	
	CSTCE10M0G52-R0	SMD	10.0	内部 (10)	内部 (10)	1.8	

注意事项　上述振荡常数是振荡器生产商在特定环境下评估得出的参考值。如果在实际应用中必须对振荡器特性进行优化，请向振荡器生产商申请对实现电路的评估。振荡电压和振荡频率仅表示振荡器特性。使用 **78K0/LE3** 的内部工作条件不要超出 **AC** 特性和 **DC** 特性的额定值。

标准产品

DC 特性（1/5）

（T_A = –40 至+85°C，1.8 V ≤ V_{DD} ≤ 5.5 V，AV_{REF} ≤ V_{DD}，V_{SS} = AV_{SS} = 0 V）

参数	符号	条件		最小值	典型值	最大值	单位
输出电流，高[注1]	I_{OH1}	P11 至 P14，P31 至 P34，P40 至 P44，P120 的每一个引脚	4.0 V ≤ V_{DD} ≤ 5.5 V			–3.0	mA
			2.7 V ≤ V_{DD} < 4.0 V			–2.5	mA
			1.8 V ≤ V_{DD} < 2.7 V			–1.0	mA
		P80 至 P83， P100 至 P103，P110 至 P113， P140 至 P143，P150 至 P153 的每一个引脚	4.0 V ≤ V_{DD} ≤ 5.5 V			–0.1	mA
			2.7 V ≤ V_{DD} < 4.0 V			–0.1	mA
			1.8 V ≤ V_{DD} < 2.7 V			–0.1	mA
		P11 至 P14, P31 至 P34, P40 至 P44，P120 的所有引脚[注3]	4.0 V ≤ V_{DD} ≤ 5.5 V			–20.0	mA
			2.7 V ≤ V_{DD} < 4.0 V			–10.0	mA
			1.8 V ≤ V_{DD} < 2.7 V			–5.0	mA
		P80 至 P83， P100 至 P103，P110 至 P113， P140 至 P143，P150 至 P153 的所有引脚[注3]	4.0 V ≤ V_{DD} ≤ 5.5 V			–2.8	mA
			2.7 V ≤ V_{DD} < 4.0 V			–2.8	mA
			1.8 V ≤ V_{DD} < 2.7 V			–2.8	mA
		所有引脚的总和[注3]	4.0 V ≤ V_{DD} ≤ 5.5 V			–22.8	mA
			2.7 V ≤ V_{DD} < 4.0 V			–12.8	mA
			1.8 V ≤ V_{DD} < 2.7 V			–7.8	mA
	I_{OH2}	P20 至 P27 的每一个引脚	AV_{REF} = V_{DD}			–0.1	mA
输出电流，低[注2]	I_{OL1}	P11 至 P14，P31 至 P34，P40 至 P44，P120 的每一个引脚	4.0 V ≤ V_{DD} ≤ 5.5 V			8.5	mA
			2.7 V ≤ V_{DD} < 4.0 V			5.0	mA
			1.8 V ≤ V_{DD} < 2.7 V			2.0	mA
		P80 至 P83， P100 至 P103，P110 至 P113， P140 至 P143，P150 至 P153 的每一个引脚	4.0 V ≤ V_{DD} ≤ 5.5 V			0.4	mA
			2.7 V ≤ V_{DD} < 4.0 V			0.4	mA
			1.8 V ≤ V_{DD} < 2.7 V			0.4	mA
		P11 至 P14, P31 至 P34, P40 至 P44，P120 的所有引脚[注3]	4.0 V ≤ V_{DD} ≤ 5.5 V			20.0	mA
			2.7 V ≤ V_{DD} < 4.0 V			15.0	mA
			1.8 V ≤ V_{DD} < 2.7 V			9.0	mA
		P80 至 P83， P100 至 P103，P110 至 P113， P140 至 P143，P150 至 P153 的所有引脚[注3]	4.0 V ≤ V_{DD} ≤ 5.5 V			11.2	mA
			2.7 V ≤ V_{DD} < 4.0 V			11.2	mA
			1.8 V ≤ V_{DD} < 2.7 V			11.2	mA
		所有引脚的总和[注3]	4.0 V ≤ V_{DD} ≤ 5.5 V			31.2	mA
			2.7 V ≤ V_{DD} < 4.0 V			26.2	mA
			1.8 V ≤ V_{DD} < 2.7 V			20.2	mA
	I_{OL2}	P20 至 P27 的每一个引脚	AV_{REF} = V_{DD}			0.4	mA

注　**1.** 即使电流从 V_{DD} 流向输出引脚，设备操作时的电流值也可以得到保证。

2. 即使电流从输出引脚流向 GND，设备操作时的电流值也可以得到保证。

3. 占空比=70%条件下（输出电流的时间为 0.7 × t，不输出电流的时间为 0.3 × t，其中 t 为特定时间）的规范。占空比不等于 70%时，可以通过以下表达式计算引脚输出电流的总和。

- 当 I_{OH} 的占空比为 n%：引脚总的输出电流 = （I_{OH} × 0.7）/（n × 0.01）

<举例>　当占空比为 50%，I_{OH} = 20.0 mA

引脚总的输出电流 = （20.0 × 0.7）/（50 × 0.01） = 28.0 mA

但是，允许流入每个引脚的电流不会随占空比而变化。高于最大额定值的电流绝对不能流入引脚。

备注　除非另外说明，复用功能引脚的特性与端口引脚的相同。

标准产品

DC 特性（2/5）

（T_A = –40 至+85°C，1.8 V ≤ V_{DD} ≤ 5.5 V，AV_{REF} ≤ V_{DD}，V_{SS} = AV_{SS} = 0 V）

参数	符号	条件		最小值	典型值	最大值	单位
输入电压，高	V_{IH1}	P10，P16，P17，P32，P80 至 P83，P100 至 P103，P110 至 P112，P121 至 P124，P140 至 P143，P150 至 P153		0.7V_{DD}		V_{DD}	V
	V_{IH2}	P11 至 P15，P30，P31，P33，P34，P40 至 P44，P113，P120，$\overline{RESET}$，EXCLK		0.8V_{DD}		V_{DD}	V
	V_{IH3}	P20 至 P27	AV_{REF} = V_{DD}	0.7AV_{REF}		AV_{REF}	V
输入电压，低	V_{IL1}	P10，P16，P17，P32，P80 至 P83，P100 至 P103，P110 至 P112，P121 至 P124，P140 至 P143，P150 至 P153		0		0.3V_{DD}	V
	V_{IL2}	P11 至 P15，P30，P31，P33，P34，P40 至 P44，P113，P120，$\overline{RESET}$，EXCLK		0		0.2V_{DD}	V
	V_{IL3}	P20 至 P27	AV_{REF} = V_{DD}	0		0.3AV_{REF}	V
输出电压，高	V_{OH1}	P11 至 P14，P31 至 P34，P40 至 P44，P120	4.0 V ≤ V_{DD} ≤ 5.5 V，I_{OH1} = –3.0 mA	V_{DD} – 0.7			V
			2.7 V ≤ V_{DD} < 4.0 V，I_{OH1} = –2.5 mA	V_{DD} – 0.5			V
			1.8 V ≤ V_{DD} < 2.7 V，I_{OH1} = –1.0 mA	V_{DD} – 0.5			V
		P80 至 P83，P100 至 P103，P110 至 P113，P140 至 P143，P150 至 P153	I_{OH1} = –0.1 mA	V_{DD} – 0.5			V
	V_{OH2}	P20 至 P27	AV_{REF} = V_{DD}，I_{OH2} = –0.1 mA	V_{DD} – 0.5			V
输出电压，低	V_{OL1}	P11 至 P14，P31 至 P34，P40 至 P44，P120	4.0 V ≤ V_{DD} ≤ 5.5 V，I_{OL1} = 8.5 mA			0.7	V
			2.7 V ≤ V_{DD} < 4.0 V，I_{OL1} = 5.0 mA			0.7	V
			1.8 V ≤ V_{DD} < 2.7 V，I_{OL1} = 2.0 mA			0.5	V
			1.8 V ≤ V_{DD} < 2.7 V，I_{OL1} = 1.0 mA			0.5	V
			1.8 V ≤ V_{DD} < 2.7 V，I_{OL1} = 0.5 mA			0.4	V
		P80 至 P83，P100 至 P103，P110 至 P113，P140 至 P143，P150 至 P153	I_{OL1} = 0.4 mA			0.4	V
	V_{OL2}	P20 至 P27	AV_{REF} = V_{DD}，I_{OL2} = 0.4 mA			0.4	V

备注　除非另外说明，复用功能引脚的特性与端口引脚的相同。

注意事项　P122/EXCLK 的高电平和低电平输入电压在输入端口模式和外部时钟模式时会有变化。

标准产品

DC 特性（3/5）

（T_A = –40 至+85°C，1.8 V ≤ V_{DD} ≤ 5.5 V，AV_{REF} ≤ V_{DD}，V_{SS} = AV_{SS} = 0 V）

参数	符号	条件			最小值	典型值	最大值	单位
输入漏电流，高	I_{LIH1}	P11 至 P14, P31 至 P34, P40 至 P44, P80 至 P83, P100 至 P103, P110 至 P113, P120, P140 至 P143, P150 至 P153, FLMD0，$\overline{\text{RESET}}$	V_I = V_{DD}				1	μA
	I_{LIH2}	P20 至 P27	V_I = AV_{REF} = V_{DD}				1	μA
	I_{LIH3}	P121 至 124 （X1，X2，XT1，XT2）	V_I = V_{DD}	I/O 端口模式			1	μA
				OSC 模式			20	μA
输入漏电流，低	I_{LIL1}	P11 至 P14, P31 至 P34, P40 至 P44, P80 至 P83, P100 至 P103, P110 至 P113, P120, P140 至 P143, P150 至 P153, FLMD0，$\overline{\text{RESET}}$	V_I = V_{SS}				–1	μA
	I_{LIL2}	P20 至 P27	V_I = V_{SS}，AV_{REF} = V_{DD}				–1	μA
	I_{LIL3}	P121 至 124 （X1，X2，XT1，XT2）	V_I = V_{SS}	I/O 端口模式			–1	μA
				OSC 模式			–20	μA
上拉电阻	R_U	V_I = V_{SS}			10	20	100	kΩ
FLMD0 电源电压	V_{IL}	正常操作模式下			0		0.2V_{DD}	V
	V_{IH}	自编程模式下			0.8V_{DD}		V_{DD}	V

备注　　除非另外说明，复用功能引脚的特性与端口引脚的相同。

标准产品

DC 特性（4/5）

（T_A = –40 至+85°C，1.8 V ≤ V_{DD} ≤ 5.5 V，AV_{REF} ≤ V_{DD}，V_{SS} = AV_{SS} = 0 V）

参数	符号	条件			最小值	典型值	最大值	单位
供电电流[注 1]	I_{DD1}	操作模式	f_{XH} = 10 MHz [注 2], V_{DD} = 5.0 V	方波输入		1.6	3.0	mA
				振荡器连接		2.3	3.4	
			f_{XH} = 10 MHz [注 2], V_{DD} = 3.0 V	方波输入		1.5	2.9	mA
				振荡器连接		2.2	3.3	
			f_{XH} = 5 MHz [注 2], V_{DD} = 3.0 V	方波输入		0.9	1.7	mA
				振荡器连接		1.3	2.0	
			f_{XH} = 5 MHz [注 2], V_{DD} = 2.0 V	方波输入		0.7	1.4	mA
				振荡器连接		1.0	1.6	
			f_{RH} = 8 MHz，V_{DD} = 5.0 V [注 3]			1.4	2.3	mA
			f_{SUB} = 32.768 kHz [注 4], V_{DD} = 5.0 V	振荡器连接		6.7	26	μA
	I_{DD2}	HALT 模式	f_{XH} = 10 MHz [注 2], V_{DD} = 5.0 V	方波输入		0.4	1.4	mA
				振荡器连接		1.0	1.7	
			f_{XH} = 5 MHz [注 2], V_{DD} = 3.0 V	方波输入		0.2	0.7	mA
				振荡器连接		0.5	1.0	
			f_{RH} = 8 MHz，V_{DD} = 5.0 V [注 3]			0.4	1.2	mA
			f_{SUB} = 32.768 kHz [注 4], V_{DD} = 5.0 V	振荡器连接		2.4	22	μA
	I_{DD3} [注 5]	STOP 模式	V_{DD} = 5.0 V			1	20	μA
			V_{DD} = 5.0 V，T_A = –40 至+70°C			1	10	μA

<R>

注　**1.** 流入内部供电电源的总电流（V_{DD}），包括当输入引脚固定为 V_{DD} 或 V_{SS} 时的输入漏电流。最大值外部操作电流。但是，不包括流入上拉电阻的电流和端口的输出电流。

2. 不包括 8 MHz 内部振荡器、240 kHz 内部振荡器和 XT1 振荡的操作电流，也不包括流入 A/D 转换器、看门狗定时器、LVI 电路和 LCD 控制器/驱动器的电流。

3. 不包括 X1 振荡、XT1 振荡和 240 kHz 内部振荡器的操作电流，也不包括流入 A/D 转换器、看门狗定时器、LVI 电路和 LCD 控制器/驱动器的电流。

4. 不包括 X1 振荡、8 MHz 内部振荡器和 240 kHz 内部振荡器的操作电流，也不包括流入 A/D 转换器、看门狗定时器、LVI 电路和 LCD 控制器/驱动器的电流。

<R> **5.** 不包括 X1 振荡、8 MHz 内部振荡器和 240 kHz 内部振荡器的操作电流，也不包括流入 A/D 转换器、看门狗定时器、LVI 电路、LCD 控制器/驱动器和实时计数器的电流。

<R> **6.** 流入内部供电电源的总电流（V_{DD}），包括当输入引脚固定为 V_{DD} 或 V_{SS} 时的输入漏电流。最大值外部操作电流。但是，不包括流入上拉电阻的电流和端口的输出电流，不包括 240 kHz 内部振荡器和 XT1 振荡的操作电流，也不包括流入 A/D 转换器、看门狗定时器、LVI 电路、LCD 控制器/驱动器和实时计数器的电流。

备注　**1.** f_{XH}：　高速系统时钟频率（X1 时钟振荡频率或外部主系统时钟频率）

2. f_{RH}：　内部高速振荡时钟频率

3. f_{SUB}：　副系统时钟频率　（XT1 时钟振荡频）

标准产品

DC 特性（5/5）

（T_A = –40 至+85°C，1.8 V ≤ V_{DD} ≤ 5.5 V，AV_{REF} ≤ V_{DD}，V_{SS} = AV_{SS} = 0 V）

	参数	符号	条件		最小值	典型值	最大值	单位
	看门狗定时器操作电流	I_{WDT}[注 1]	240 kHz 内部低速振荡时钟操作期间			5	10	μA
	LVI 操作电流	I_{LVI}[注 2]				9	18	μA
	逐次逼近型 A/D 转换器操作电流	I_{ADC1}[注 3]		2.3 V ≤ AV_{REF} ≤ V_{DD}		0.86	1.9	mA
	ΔΣ 型 A/D 转换器操作电流	I_{ADC2}[注 3]		2.7 V ≤ AV_{REF} ≤ V_{DD}		1.4	2.7	mA
<R>	LCD 操作电流	I_{LCD1}[注 4]	LCD 显示关闭（LCDON = 0，SCOC = 1）	V_{DD} = 5.0 V		3.0	8.0	μA
				V_{DD} = 3.0 V		2.0	5.0	μA
		I_{LCD2}[注 4]	LCD 显示打开（LCDON = 1，SCOC = 1）	V_{DD} = 5.0 V		3.0	8.0	μA
				V_{DD} = 3.0 V		2.0	5.0	μA

注

1. 仅包括流过看门狗定时器（包括 240 kHz 内部振荡器的操作电流）的电流。当在 HALT 模式或 STOP 模式下操作看门狗定时器时，78K0/LE3 的电流值是 I_{DD2} 或 I_{DD3} 加上 I_{WDT}。
2. 仅包括流过 LVI 电路的电流。当在 HALT 模式或 STOP 模式下操作 LVI 电路时，78K0/LE3 的电流值是 I_{DD2} 或 I_{DD3} 加上 I_{LVI}。
3. 仅包括流过 A/D 转换器（AV_{REF}）的电流。当在 HALT 模式或 STOP 模式下操作 A/D 转换器时，78K0/LE3 的电流值是 I_{DD1} 或 I_{DD2} 加上 I_{ADC1} 或 I_{ADC2}。
4. 仅包括流过 LCD 控制器/驱动器的电流。不包括流过 LCD 分压电阻的电流。78K0/LE3 的电流值是供电电流（I_{DD1}，I_{DD2} 或 I_{DD3}）加上 LCD 工作电流（I_{LCD1} 或 I_{LCD2}）。

AC 特性

（1） 基本操作

（T_A = –40 至+85°C，1.8 V ≤ V_{DD} ≤ 5.5 V，V_{SS} = AV_{SS} = 0 V）

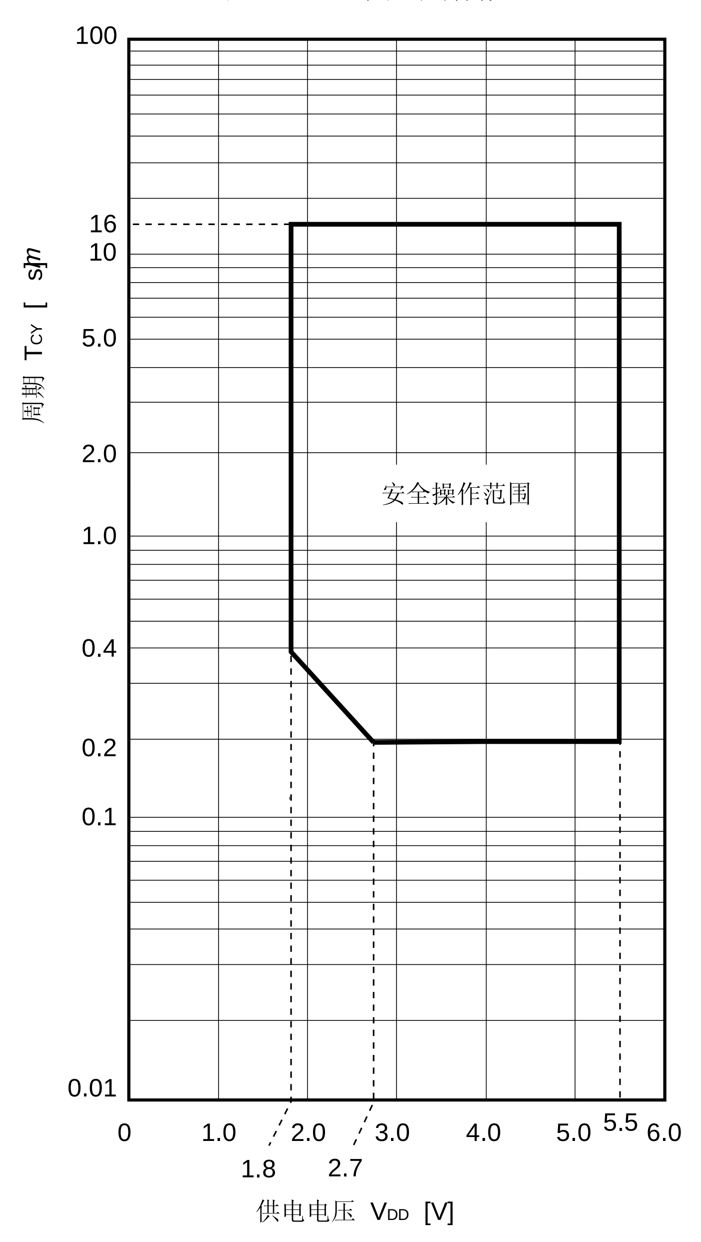
参数	符号	条件		最小值	典型值	最大值	单位
指令周期（最短指令执行时间）	T_{CY}	主系统时钟（f_{XP}）运行	2.7 V ≤ V_{DD} ≤ 5.5 V	0.2		16	μs
			1.8 V ≤ V_{DD} < 2.7 V	0.4		16	μs
		副系统时钟（f_{SUB}）运行		114	122	125	μs
外设硬件时钟频率	f_{PNS}	XSEL = 1	2.7 V ≤ V_{DD} ≤ 5.5 V			10	MHz
			1.8 V ≤ V_{DD} < 2.7 V			5	MHz
		XSEL = 0	2.7 V ≤ V_{DD} ≤ 5.5 V	7.6		8.4	MHz
			1.8 V ≤ V_{DD} < 2.7 V [注 1]	6.75		8.4	MHz
外部主系统时钟频率	f_{EXCLK}	2.7 V ≤ V_{DD} ≤ 5.5 V		2.0		10.0	MHz
		1.8 V ≤ V_{DD} < 2.7 V		2.0		5.0	MHz
外部主系统时钟输入高电平宽度，低电平宽度	t_{EXCLKH}, t_{EXCLKL}	2.7 V ≤ V_{DD} ≤ 5.5 V		48		500	ns
		1.8 V ≤ V_{DD} < 2.7 V		96		500	ns
TI000 输入高电平宽度，低电平宽度	t_{TIH0}, t_{TIL0}	2.7 V ≤ V_{DD} ≤ 5.5 V		$2/f_{sam}$ + 0.2 [注 2]			μs
		1.8 V ≤ V_{DD} < 2.7 V		$2/f_{sam}$ + 0.5 [注 2]			μs
TI50，TI51，TI52 输入频率	f_{TI5}	4.0 V ≤ V_{DD} ≤ 5.5 V	TI50，TI51			10	MHz
			TI52			16	MHz
		2.7 V ≤ V_{DD} < 4.0 V				10	MHz
		1.8 V ≤ V_{DD} < 2.7 V				5	MHz
TI50，TI51，TI52 输入高电平宽度，低电平宽度	t_{TIH5}, t_{TIL5}	4.0 V ≤ V_{DD} ≤ 5.5 V	TI50，TI51	50			ns
			TI52	31.25			ns
		2.7 V ≤ V_{DD} < 4.0 V		50			ns
		1.8 V ≤ V_{DD} < 2.7 V		100			ns
中断输入高电平宽度，低电平宽度	t_{INTH}, t_{INTL}			1			μs
按键返回输入低电平宽度	t_{KR}			250			ns
$\overline{\text{RESET}}$ 低电平宽度	t_{RSL}			10			μs

注 **1.** 主系统时钟频率特性。用外设功能将时钟分频器设置为 $f_{RH}/2$ 或更小。

2. 使用分频模式寄存器 00（PRM00）的第 0 位和第 1 位（PRM000，PRM001）可以选择选择 $f_{sam} = f_{PRS}$，$f_{PRS}/4$，$f_{PRS}/256$。注意当选择 TI000 有效沿作为计数时钟时，$f_{sam} = f_{PRS}$。

标准产品

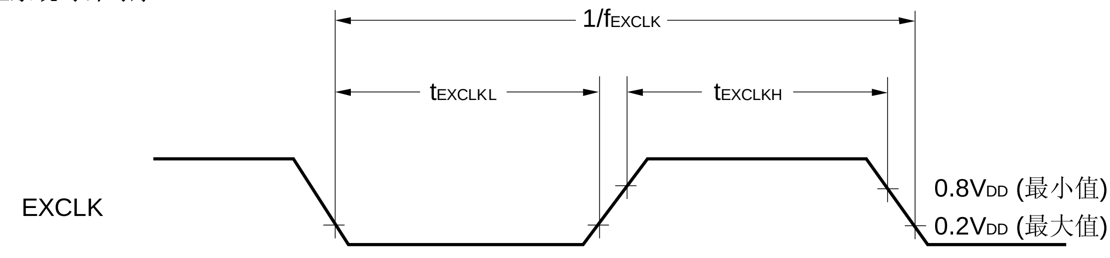
T_{CY} 和 V_{DD}（主系统时钟操作）



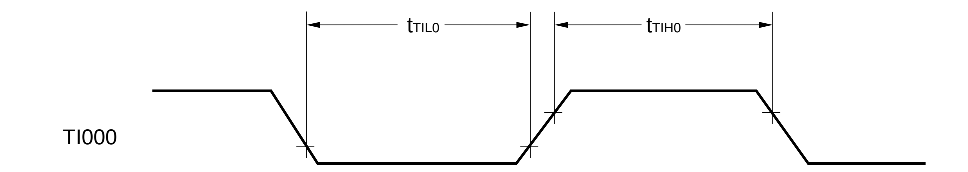
AC 时序测试点（不包括外部主系统时钟）

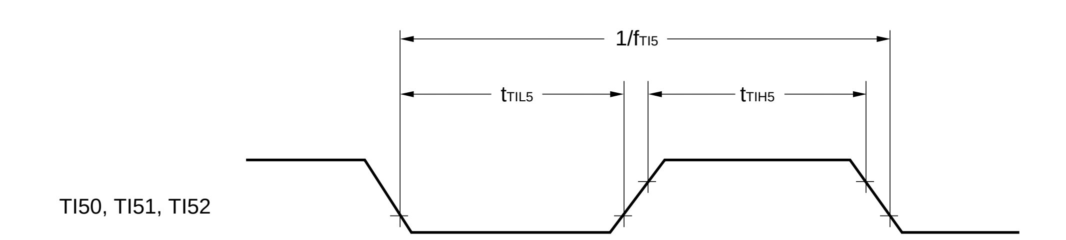


外部主系统时钟时序

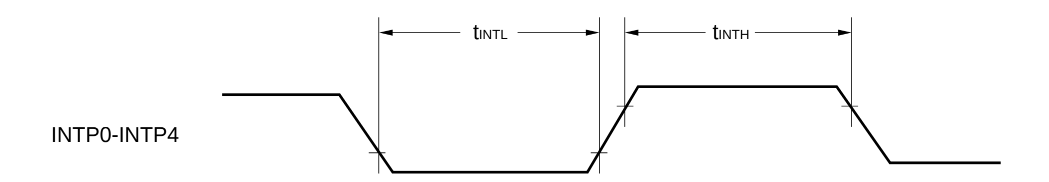


TI 时序

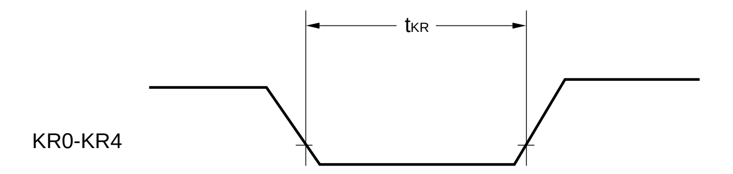




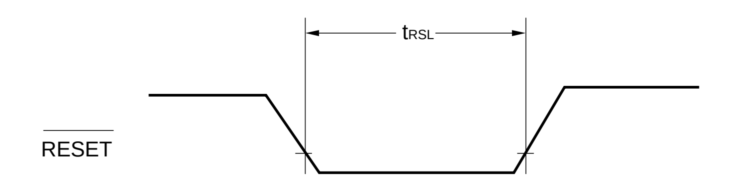
中断请求输入时序



按键中断输入时序



$\overline{\text{RESET}}$ 输入时序



（2）　串行接口

（T_A = –40 至+85°C，1.8 V ≤ V_{DD} ≤ 5.5 V，V_{SS} = AV_{SS} = 0 V）

参数	符号	条件	最小值	典型值	最大值	单位
传输速率					250	kbps

（3）　串行接口

（T_A = –40 至+85°C，1.8 V ≤ V_{DD} ≤ 5.5 V，V_{SS} = AV_{SS} = 0 V）

（a）UART6（专用波特率发生器输出）

参数	符号	条件	最小值	典型值	最大值	单位
传输速率					625	kbps

（b）UART0（专用波特率发生器输出）

参数	符号	条件	最小值	典型值	最大值	单位
传输速率					625	kbps

（c） CSI10（主设备模式，$\overline{\text{SCK10}}$... 内部时钟输出）

参数	符号	条件	最小值	典型值	最大值	单位
$\overline{\text{SCK10}}$ 周期时间	t_{KCY1}	$2.7\ V \le V_{DD} \le 5.5\ V$	250			ns
		$1.8\ V \le V_{DD} < 2.7\ V$	500			ns
$\overline{\text{SCK10}}$ 高/低电平宽度	t_{KH1}, t_{KL1}	$2.7\ V \le V_{DD} \le 5.5\ V$	$t_{KCY1}/2 - 25$ 注 1			ns
		$1.8\ V \le V_{DD} < 2.7\ V$	$t_{KCY1}/2 - 50$ 注 1			ns
SI10 建立时间（到 $\overline{\text{SCK10}}$↑）	t_{SIK1}	$2.7\ V \le V_{DD} \le 5.5\ V$	80			ns
		$1.8\ V \le V_{DD} < 2.7\ V$	170			ns
SI10 保持时间（从 $\overline{\text{SCK10}}$↑）	t_{KSI1}		30			Ns
从 $\overline{\text{SCK10}}$↓ 到 SO10 输出的延迟时间	t_{KSO1}	C = 50 pF 注 2			40	ns

注　1. 使用高速系统时钟 （f_{XH}）时的取值。

　　2. C 是 $\overline{\text{SCK10}}$ 和 SO10 输出线的负载电容。

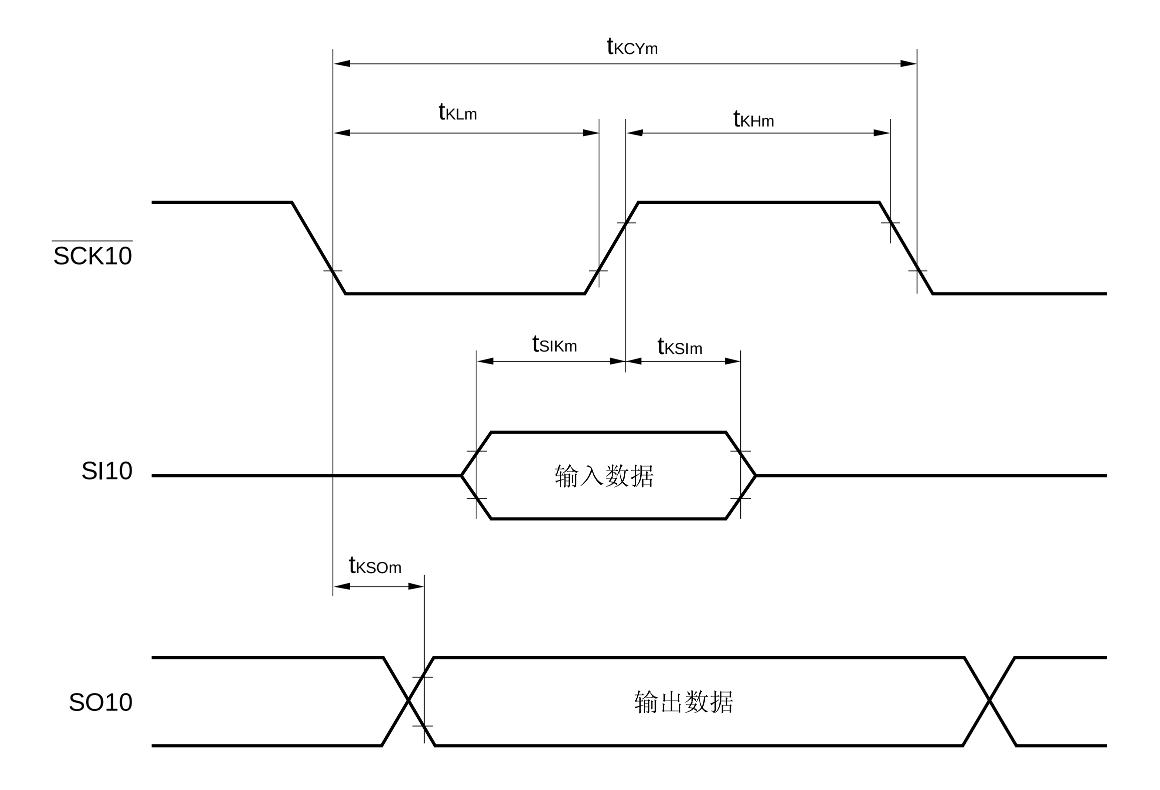
（d）CSI10（从设备模式，$\overline{\text{SCK10}}$... 外部时钟输入）

参数	符号	条件		最小值	典型值	最大值	单位
$\overline{\text{SCK10}}$ 周期时间	t_{KCY2}			400			ns
$\overline{\text{SCK10}}$ 高/低电平宽度	t_{KH2}, t_{KL2}			$t_{KCY2}/2$			ns
SI10 建立时间（到 $\overline{\text{SCK10}}$↑）	t_{SIK2}			80			ns
SI10 保持时间（从 $\overline{\text{SCK10}}$↑）	t_{KSI2}			50			ns
从 $\overline{\text{SCK10}}$↓ 到 SO10 输出的延迟时间	t_{KSO2}	C = 50 pF 注	$2.7\ V \le V_{DD} \le 5.5\ V$			120	ns
			$1.8\ V \le V_{DD} < 2.7\ V$			165	ns

注　C 是 SO10 输出线的负载电容。

串行发送时序

CSI10：



备注　　m = 1，2

10 位逐次逼近型 A/D 转换器特性（仅限μPD78F045x 和 78F046x）

（T_A = –40 至+85°C，2.3 V ≤ AV_{REF} ≤ V_{DD} ≤ 5.5 V，V_{SS} = AV_{SS} = 0 V）

参数	符号	条件	最小值	典型值	最大值	单位
分辨率	R_{ES1}				10	位
总误差[注 1，2]	A_{INL}	4.0 V ≤ AV_{REF} ≤ 5.5 V			±0.4	%FSR
		2.7 V ≤ AV_{REF} < 4.0 V			±0.6	%FSR
		2.3 V ≤ AV_{REF} < 2.7 V			±1.2	%FSR
转换时间	t_{CONV}	4.0 V ≤ AV_{REF} ≤ 5.5 V	6.1		36.7	μs
		2.7 V ≤ AV_{REF} < 4.0 V	12.2		36.7	μs
		2.3 V ≤ AV_{REF} < 2.7 V	27		66.6	μs
零标度误差[注 1，2]	E_{ZS}	4.0 V ≤ AV_{REF} ≤ 5.5 V			±0.4	%FSR
		2.7 V ≤ AV_{REF} < 4.0 V			±0.6	%FSR
		2.3 V ≤ AV_{REF} < 2.7 V			±0.6	%FSR
满量程误差[注 1，2]	E_{FS}	4.0 V ≤ AV_{REF} ≤ 5.5 V			±0.4	%FSR
		2.7 V ≤ AV_{REF} < 4.0 V			±0.6	%FSR
		2.3 V ≤ AV_{REF} < 2.7 V			±0.6	%FSR
积分非线性误差[注 1]	I_{LE1}	4.0 V ≤ AV_{REF} ≤ 5.5 V			±2.5	LSB
		2.7 V ≤ AV_{REF} < 4.0 V			±4.5	LSB
		2.3 V ≤ AV_{REF} < 2.7 V			±6.5	LSB
微分非线性误差[注 1]	D_{LE1}	4.0 V ≤ AV_{REF} ≤ 5.5 V			±1.5	LSB
		2.7 V ≤ AV_{REF} < 4.0 V			±2.0	LSB
		2.3 V ≤ AV_{REF} < 2.7 V			±2.0	LSB
模拟输入电压	V_{AIN1}		AV_{SS}		AV_{REF}	V

注　**1.** 不包括量化误差（±1/2 LSB）

　　2. 该值表示满量程的比率（%FSR）

<R> **16 位 ΔΣ 型 A/D 转换器特性（仅限μPD78F046x）**

（T_A = –40 至+85°C，2.7 V ≤ AV_{REF} ≤ V_{DD} ≤ 5.5 V，V_{SS} = AV_{SS} = 0 V）

参数	符号	条件			最小值	典型值	最大值	单位
分辨率	R_{ES2}				8		16	位
采样时钟[注 1]	f_{VP}	差分输入时		3.5 V ≤ AV_{REF} ≤ 5.5 V	0.016		1.25	MHz
				2.7 V ≤ AV_{REF} < 3.5 V	0.016		0.625	MHz
		单信号输入时		2.85 V ≤ AV_{REF} ≤ 5.5 V	0.016		0.625	MHz
				2.7 V ≤ AV_{REF} < 2.85 V	0.016		0.525	MHz
积分非线性误差（相对准确度）	I_{LE2}	差分输入时[注 2]	14 位分辨率[注 3]	AV_{REF} = 5.0 V		±1.0		LSB
				3.5 V ≤ AV_{REF} ≤ 5.5 V		±1.7		LSB
				2.7 V ≤ AV_{REF} < 3.5 V		±2.6		LSB
		单信号输入时[注 2]	12 位分辨率[注 3]			±2.8		LSB
微分非线性误差（相对准确度）	D_{LE2}	差分输入时[注 2]	14 位分辨率[注 3]	AV_{REF} = 5.0 V		±1.0		LSB
				3.5 V ≤ AV_{REF} ≤ 5.5 V		±1.7		LSB
				2.7 V ≤ AV_{REF} < 3.5 V		±2.6		LSB
		单信号输入时[注 2]	12 位分辨率[注 3]			±2.8		LSB
偏移量	EOS	差分输入时				±0.032		%FSR
		单信号输入时				±0.16		%FSR
增益误差	GE	差分输入时				±0.09		%
		单信号输入时				±0.1		%
参考电压	REF+					AV_{REF}		V
	REF–					AV_{SS}		V
模拟输入电压	V_{AIN2}	高精度模式关闭			0		REF+	V
		高精度模式打开			0.1REF+		0.9REF+	V

注 **1.** 可以通过下列表达式计算转换时间，基于采样时钟(f_{VP}) 并设置分辨率 (N 位)。

转换时间 = $2^N / f_{VP}$

2. 在差分输入期间高精度模式被设置为打开时，或者在单信号输入期间高精度模式被设置为关闭时，适用这些值。

3. 上述以外的分辨率(N 位)作为条件时，积分线性误差 (I_{LE2}) and 微分线性误差 (D_{LE2}) 栏的特性可以使用下列表达式来计算。

- 在差分输入期间

  N 位分辨率下的 I_{LE2} = 14 位分辨率下的 $I_{LE2} \times 2^{(N-14)}$

  N 位分辨率下的 D_{LE2} = 14 位分辨率下的 $D_{LE2} \times 2^{(N-14)}$

- 在单信号输入期间

  N 位分辨率下的 I_{LE2} = 12 位分辨率下的 $I_{LE2} \times 2^{(N-12)}$

  N 位分辨率下的 D_{LE2} = 12 位分辨率下的 $D_{LE2} \times 2^{(N-12)}$

备注 在 16 位 ΔΣ 型 A/D 转换器特性中，近似线通过最小二乘法来定义。

LCD 特性

（1）电阻分压方式

（a）静态显示模式（T_A = –40 至+85°C，1.8 V ≤ V_{LCD} ≤ V_{DD} ≤ 5.5 V，V_{SS} = 0 V）[注 3]

参数	符号	条件	最小值	典型值	最大值	单位
LCD 驱动电压	V_{LCD}	注 3			V_{DD}	V
LCD 分压电阻[注 1]	R_{LCD}		60	100	150	kΩ
LCD 输出电阻[注 2]（公共端）	R_{ODC}				40	kΩ
LCD 输出电阻[注 2]（Segment）	R_{ODS}				200	kΩ

（b） 1/3 偏压方式（T_A = –40 至+85°C，1.8 V ≤ V_{LCD} ≤ V_{DD} ≤ 5.5 V，V_{SS} = 0 V）[注 3]

参数	符号	条件	最小值	典型值	最大值	单位
LCD 驱动电压	V_{LCD}	注 3			V_{DD}	V
LCD 分压电阻[注 1]	R_{LCD}		60	100	150	kΩ
LCD 输出电阻[注 2]（公共端）	R_{ODC}				40	kΩ
LCD 输出电阻[注 2]（Segment）	R_{ODS}				200	kΩ

（c）1/2 偏压方式，1/4 偏压方式（T_A = –40 至+85°C，4.5 V ≤ V_{LCD} ≤ V_{DD} ≤ 5.5 V，V_{SS} = 0 V）[注 3]

参数	符号	条件	最小值	典型值	最大值	单位
LCD 驱动电压	V_{LCD}	注 3			V_{DD}	V
LCD 分压电阻[注 1]	R_{LCD}		60	100	150	kΩ
LCD 输出电阻[注 2]（公共端）	R_{ODC}				40	kΩ
LCD 输出电阻[注 2]（Segment）	R_{ODS}				200	kΩ

注　**1.** 仅限内部分压电阻方式。

2. 输出电阻是指连接在 V_{LC0}，V_{LC1}，V_{LC2} 和 V_{SS} 其中一个引脚和 SEG 或 COM 引脚之间的电阻。

3. 按照下列条件设置 VAON 。

<当设置为静态模式时>

- 当 2.0V≤ V_{LCD} ≤ V_{DD} ≤ 5.5 V 时：VAON = 0
- 当 1.8V≤ V_{LCD} ≤ V_{DD} ≤ 3.6 V 时：VAON = 1

<当设置为 1/3 偏压模式时>

- 当 2.5V≤ V_{LCD} ≤ V_{DD} ≤ 5.5 V 时：VAON = 0
- 当 1.8V≤ V_{LCD} ≤ V_{DD} ≤ 3.6 V 时：VAON = 1

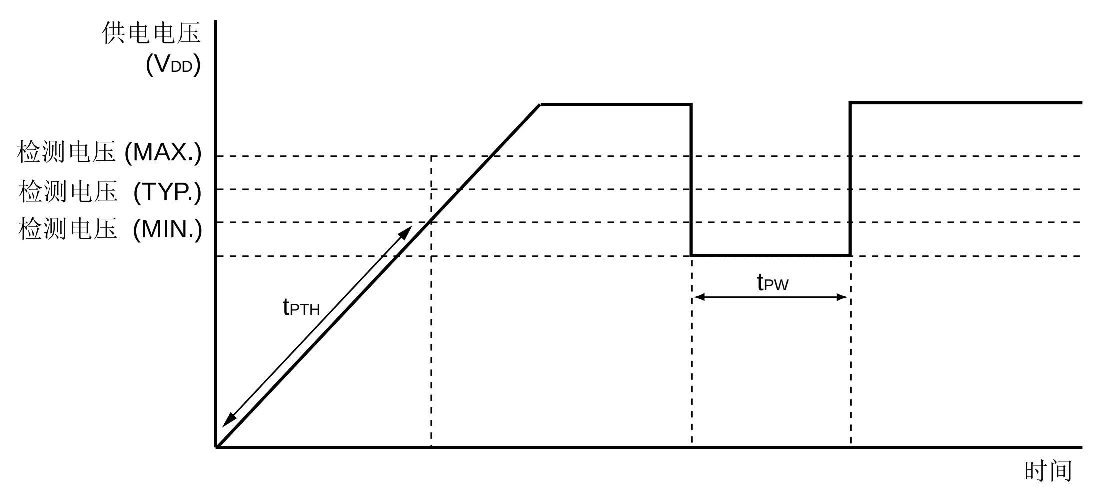
<R> <当设置为 1/2 偏压模式或 1/4 偏压模式时>

- 当 2.7V≤ V_{LCD} ≤ V_{DD} ≤ 5.5 V 时：VAON = 0
- 当 1.8V≤ V_{LCD} ≤ V_{DD} ≤ 3.6 V 时：VAON = 1

1.59 V POC 电路特性（T_A = –40 至+85°C，V_{SS} = 0 V）

参数	符号	条件	最小值	典型值	最大值	单位
检测电压	V_{POC}		1.44	1.59	1.74	V
供电电压上升斜率	t_{PTH}	V_{DD}：0 V → V_{POC} 变化斜率	0.5			V/ms
最小脉宽	t_{PW}		200			μs

POC 电路时序

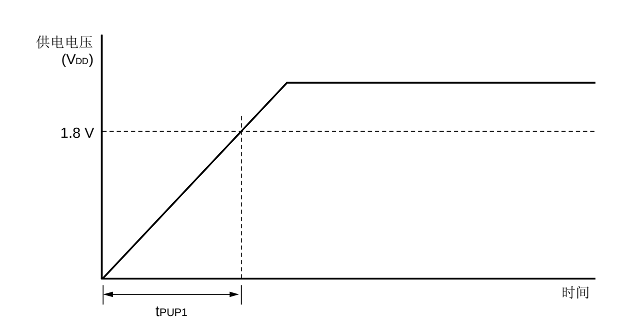


供电电压上升时间（T_A = –40 至+85°C，V_{SS} = 0 V）

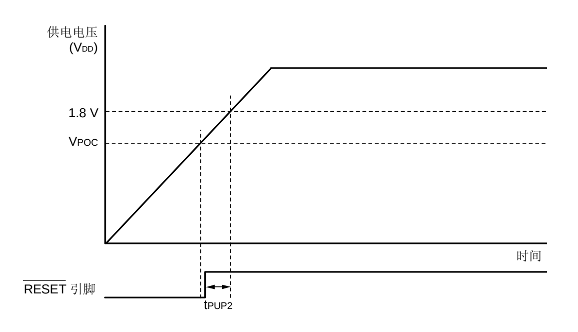
参数	符号	条件	最小值	典型值	最大值	单位
上升至 1.8 V （V_{DD} （最小值））所需的最长时间（V_{DD}: 0 V → 1.8 V）	t_{PUP1}	当没有使用 $\overline{RESET}$ 输入时，POCMODE （选项字节） = 0			3.6	ms
上升至 1.8 V （V_{DD} （最小值））所需的最长时间（释放 RESET 输入 → V_{DD}: 1.8 V）	t_{PUP2}	当使用 $\overline{RESET}$ 输入时，POCMODE （选项字节） = 0			1.9	ms

供电电压上升时间时序

- 当未使用$\overline{RESET}$引脚时



- 当使用$\overline{RESET}$引脚时



2.7 V POC 电路特性（T_A = –40 至+85°C，V_{SS} = 0 V）

参数	符号	条件	最小值	典型值	最大值	单位
供电电压应用中的检测电压	V_{DDPOC}	POCMODE （选项字节） = 1	2.50	2.70	2.90	V

LVI 电路特性（T_A = –40 至+85°C，$V_{POC} \le V_{DD} \le 5.5$ V，V_{SS} = 0 V）

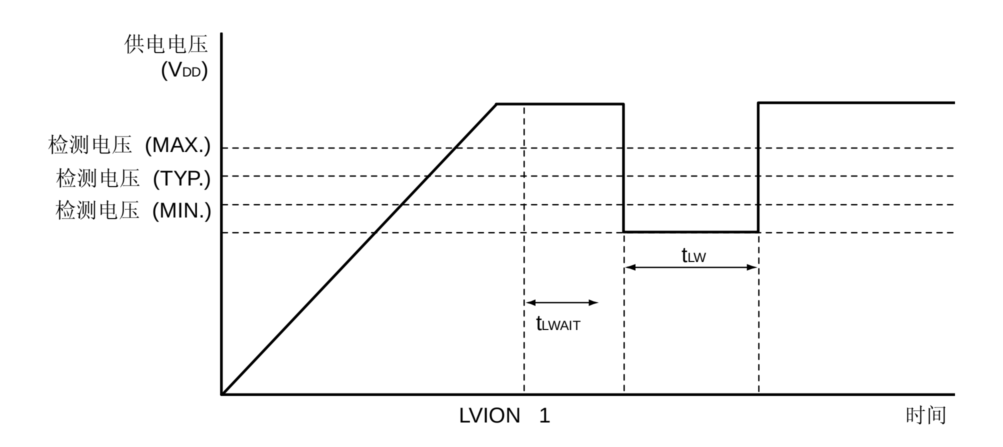
参数		符号	条件	最小值	典型值	最大值	单位
检测电压	供电电压电平	V_{LVI0}		4.14	4.24	4.34	V
		V_{LVI1}		3.99	4.09	4.19	V
		V_{LVI2}		3.83	3.93	4.03	V
		V_{LVI3}		3.68	3.78	3.88	V
		V_{LVI4}		3.52	3.62	3.72	V
		V_{LVI5}		3.37	3.47	3.57	V
		V_{LVI6}		3.22	3.32	3.42	V
		V_{LVI7}		3.06	3.16	3.26	V
		V_{LVI8}		2.91	3.01	3.11	V
		V_{LVI9}		2.75	2.85	2.95	V
		V_{LVI10}		2.60	2.70	2.80	V
		V_{LVI11}		2.45	2.55	2.65	V
		V_{LVI12}		2.29	2.39	2.49	V
		V_{LVI13}		2.14	2.24	2.34	V
		V_{LVI14}		1.98	2.08	2.18	V
		V_{LVI15}		1.83	1.93	2.03	V
	外部输入引脚注 1	EXLVI	EXLVI < V_{DD}，1.8 V ≤ V_{DD} ≤ 5.5 V	1.11	1.21	1.31	V
最短脉冲宽度				200			μs
操作稳定等待时间注 2						10	μs

注　**1.** 使用 EXLVI/P120/INTP0 引脚。

　　2. 从设置低电压检测寄存器（LVIM）的第 7 位（LVION）为 1 到操作稳定所需的时间。

备注　$V_{LVI(n-1)} > V_{LVIn}$：n = 1 至 15

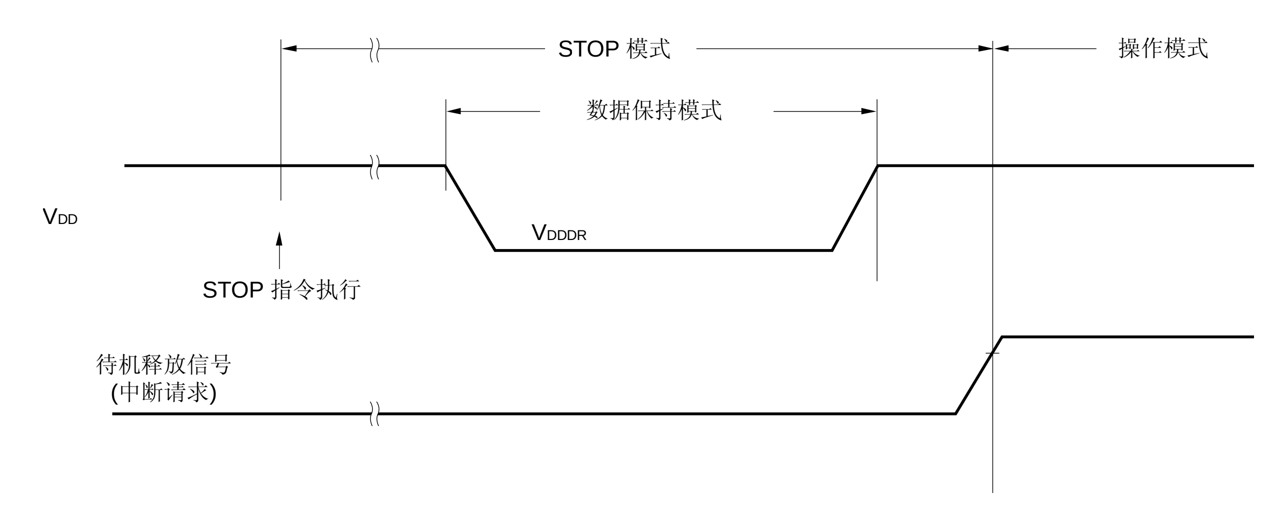
LVI 电路时序



数据存储器 STOP 模式低供电电压时的数据保持特性（T_A = –40 至+85°C）

参数	符号	条件	最小值	典型值	最大值	单位
数据保持供电电压	V_{DDDR}		1.44 [注]		5.5	V

注　该值取决于 POC 检测电压。当电压降低时，保持该数据直到 POC 复位生效，但当 POC 复位有效时，数据不能保持。



Flash 存储器编程特性
（T_A = –40 至+85°C，2.7 V ≤ V_{DD} ≤ 5.5 V，V_{SS} = AV_{SS} = 0 V）

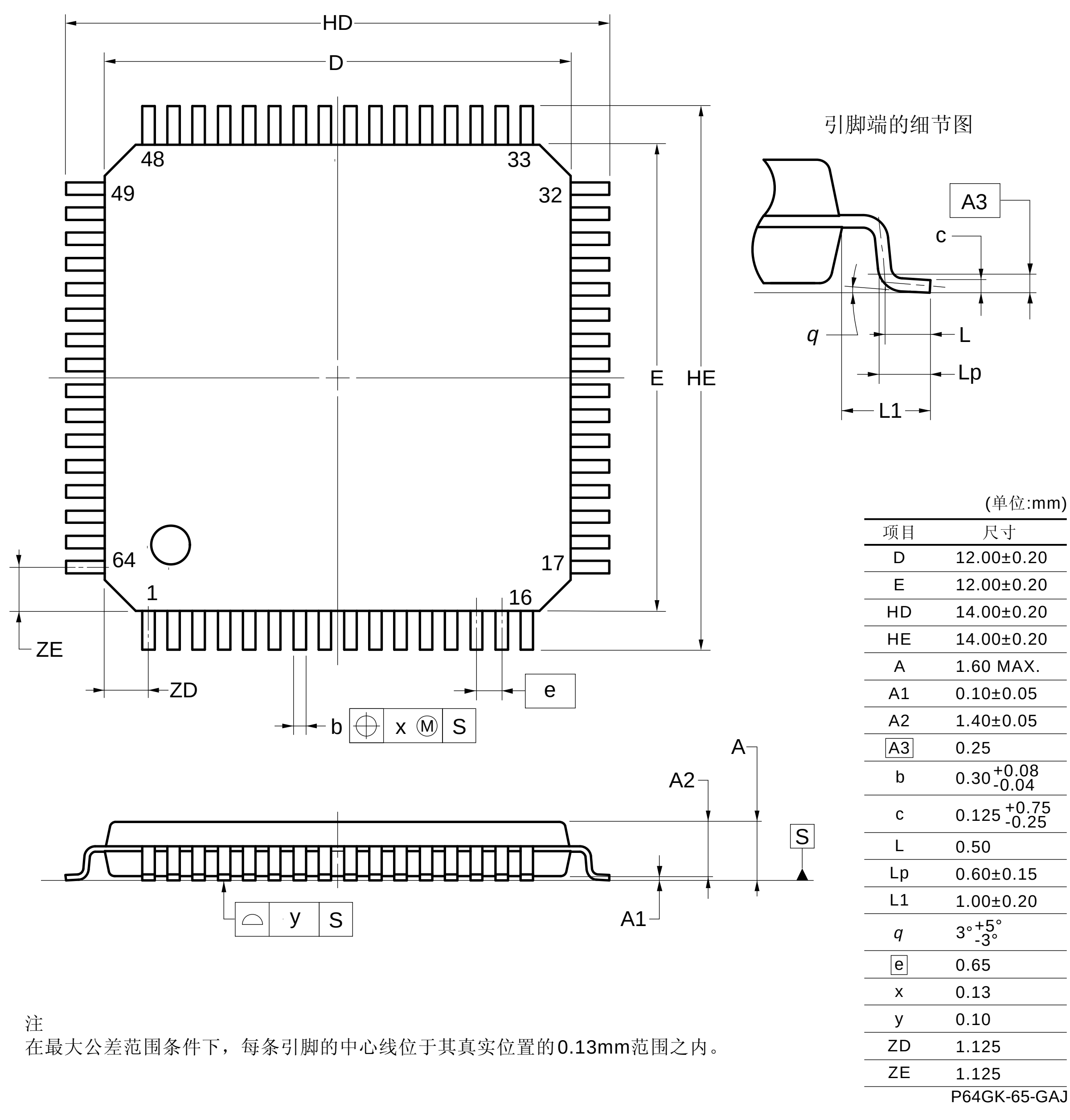
●基本特性

参数		符号	条件			最小值	典型值	最大值	单位
V_{DD} 供电电流		I_{DD}					4.5	11.0	mA
擦除时间[注 1，2]	All block	T_{eraca}					20	200	ms
	Block unit	T_{erasa}					20	200	ms
写入时间（8 位）[注 1]		T_{wrwa}					10	100	μs
每片重写次数		C_{erwr}	在擦除之后，1 次擦除 ＋1 次写入 = 1 次重写[注 3]	使用 flash 存储器编程器时，并使用 NEC 电子提供的库	保持时间：15 年	1000			次数
				使用 NEC 电子提供 EEPROM 模拟库，且可重写的 ROM 大小是 4 KB	保持时间：3 年[注 4]	10000			次数

<R> 注　**1.** Flash 存储器的特性。关于使用专用 flash 编程器 PG-FP5 时的特性，以及自编程期间的重写时间，参见**表 27-12** 和 **27-13**。
2. 不包括擦除前的预写时间和擦除验证时间（写回时间）。
3. 在出货后第一次对产品进行写操作时，"擦除 → 写"和"只写"都被作为一次重写。
<R> **4.** 数据写入之后，保证的数据保持为 3 年。如果执行重写，数据保持可以再保持此后的三年。

备注　f_{XP}：主系统时钟振荡频率

64-针塑封 LQFP (12x12)



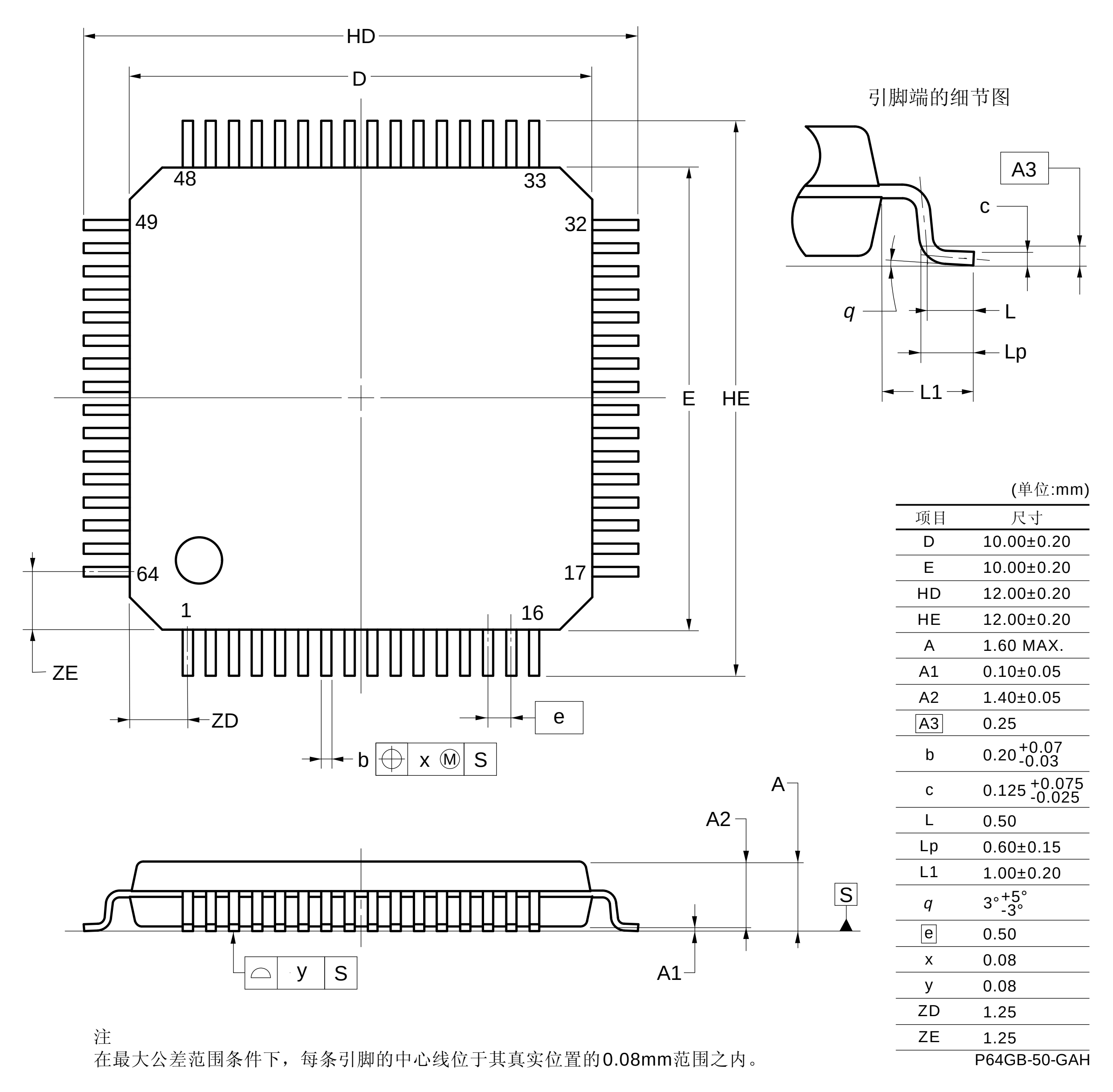
(单位:mm)

项目	尺寸
D	12.00±0.20
E	12.00±0.20
HD	14.00±0.20
HE	14.00±0.20
A	1.60 MAX.
A1	0.10±0.05
A2	1.40±0.05
A3	0.25
b	$0.30^{+0.08}_{-0.04}$
c	$0.125^{+0.75}_{-0.25}$
L	0.50
Lp	0.60±0.15
L1	1.00±0.20
q	$3°^{+5°}_{-3°}$
e	0.65
x	0.13
y	0.10
ZD	1.125
ZE	1.125

P64GK-65-GAJ

注
在最大公差范围条件下，每条引脚的中心线位于其真实位置的0.13mm范围之内。

64-针塑封 LQFP(FINE PITCH)(10x10)



(单位:mm)

项目	尺寸
D	10.00±0.20
E	10.00±0.20
HD	12.00±0.20
HE	12.00±0.20
A	1.60 MAX.
A1	0.10±0.05
A2	1.40±0.05
A3	0.25
b	$0.20^{+0.07}_{-0.03}$
c	$0.125^{+0.075}_{-0.025}$
L	0.50
Lp	0.60±0.15
L1	1.00±0.20
q	$3°^{+5°}_{-3°}$
e	0.50
x	0.08
y	0.08
ZD	1.25
ZE	1.25

P64GB-50-GAH

注
在最大公差范围条件下，每条引脚的中心线位于其真实位置的0.08mm范围之内。

<R>

第三十二章　推荐焊接条件

这些产品应该在下列推荐条件下进行焊接和安装。
如果不使用下面推荐的焊接方法和条件，请与 NEC 电子公司销售代理联系。
相关技术信息，请浏览下面的网页。

半导体设备装配手册（http://www.necel.com/pkg/en/mount/index.html）

表 32-1. 表面贴装类型焊接条件

焊接方法	焊接条件	推荐条件符号
红外回流	封装峰值温度： 260°C， 时间： 最长 60 秒 （ 大于等于 220°C）， 次数： 小于等于 3 次， 曝光限制： 7 天[注] （此后，在 125°C 预烘 10 至 72 小时）	IR60-107-3
局部加热	引脚温度： 350°C 最大值， 时间： 最长 3 秒 （每行引脚）	–

注　打开干燥包装后，在有效存放期内将其存储在 25°C 或更低温度，65% RH 或更低湿度的环境中。

注意事项　不要一起使用不同的焊接方法（局部加热法除外）。

第三十三章　等待注意事项

33.1 等待注意事项

本产品有两种内部系统总线。

一种是 CPU 总线，另一种是连接低速外设硬件的外设总线。

由于 CPU 总线时钟与外设总线时钟是异步的，因此如果对 CPU 的访问与对外设的访问发生冲突，可能会有不期望的非法数据产生。

因此当访问可能会产生冲突的外设时，CPU 反复执行处理，直至得到正确数据。

这样，CPU 并不执行下一条指令，而是在等待。如果出现这种情况，执行一条指令所需的时钟数会因为等待的时钟数而增加（等待时钟数的详细信息，参见**表 33-1**）。在执行实时操作时必须要注意这一点。

33.2 产生等待的外设硬件

表 33-1 列出了当 CPU 访问寄存器时，发出等待请求的寄存器和 CPU 等待的时钟数。

表 33-1. 产生等待的寄存器和 CPU 等待时钟数

<table>
<tr><th>外设硬件</th><th>寄存器</th><th>访问方式</th><th>等待时钟数</th></tr>
<tr><td>串行接口
UART0</td><td>ASIS0</td><td>读</td><td>1 个时钟（固定）</td></tr>
<tr><td>串行接口
UART6</td><td>ASIS6</td><td>读</td><td>1 个时钟（固定）</td></tr>
<tr><td rowspan="5">10-位逐次逼近型 A/D 转换器</td><td>ADM</td><td>写</td><td rowspan="4">1 至 5 个时钟（当选择 $f_{AD} = f_{PRS}/2$ 时）
1 至 7 个时钟（当选择 $f_{AD} = f_{PRS}/3$ 时）
1 至 9 个时钟（当选择 $f_{AD} = f_{PRS}/4$ 时）
2 至 13 个时钟（当选择 $f_{AD} = f_{PRS}/6$ 时）
2 至 17 个时钟（当选择 $f_{AD} = f_{PRS}/8$ 时）
2 至 25 个时钟（当选择 $f_{AD} = f_{PRS}/12$ 时）</td></tr>
<tr><td>ADS</td><td>写</td></tr>
<tr><td>ADPC</td><td>写</td></tr>
<tr><td>ADCR</td><td>读</td></tr>
<tr><td colspan="3">上述时钟数量是指在选择相同的源时钟作为 f_{CPU} 和 f_{PRS} 时的值。可以根据以下表达式计算下列条件时的等待时钟数量。
<计算等待时钟数量>
• 等待时钟数 =（$2f_{CPU}/f_{AD}$） + 1
* 如果等待时钟的数量≤ 0.5，则将小数点以后的部分舍去，而如果等待时钟的数量> 0.5，则只入不舍。
f_{AD}：　A/D 转换时钟的频率（$f_{PRS}/2$ 至 $f_{PRS}/12$）
f_{CPU}：　CPU 时钟频率
f_{PRS}：　外设硬件时钟频率
f_{XP}：　主系统时钟频率
<等待时钟的最大/最小数量所需的条件>
• 最大次数的数量：CPU 的最高速度（f_{XP}），A/D 转换时钟的最低速度（$f_{PRS}/12$）
• 最小次数的数量：CPU 的最低速度（$f_{SUB}/2$），A/D 转换时钟的最高速度（$f_{PRS}/2$）</td></tr>
</table>

注意事项　当 CPU 运行于副系统时钟且外设硬件时钟停止时，不要使用发生等待请求的访问方式来访问上述寄存器。

备注　　这里的时钟是指 CPU 时钟（f_{CPU}）。

附录 A 开发工具

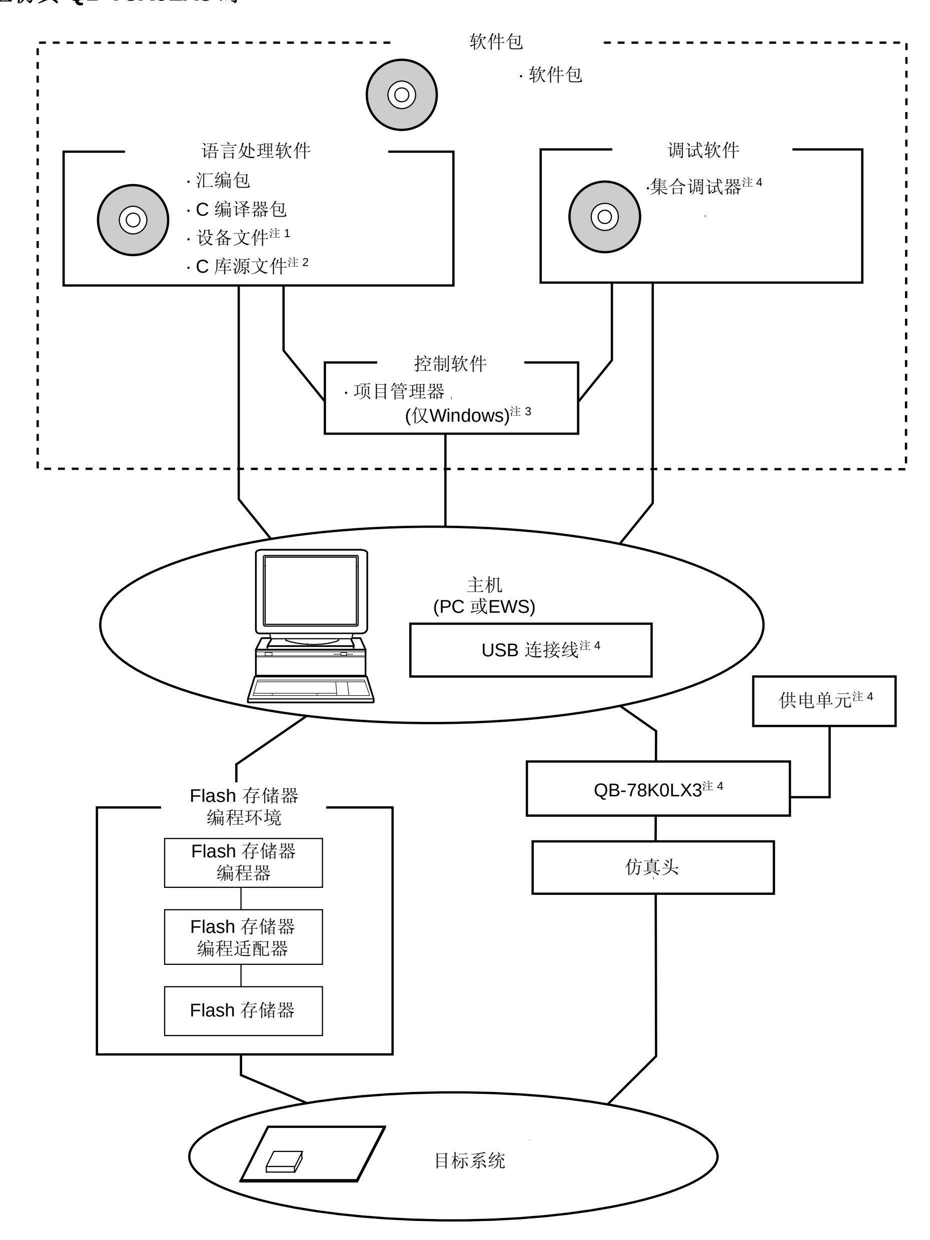
以下开发工具可用来开发使用 78K0/LE3 单片机的系统。
图 A-1 显示开发工具的配置。

- 支持 **PC98-NX** 系列
  除非另行指出，IBM PC/AT™ 支持兼容的产品兼容于 PC98-NX 系列计算机。当使用 PC98-NX 系列计算机时，可以参见 IBM PC/AT 兼容机说明。

- **WindowsTM**
  除非另行说明，“Windows”指以下操作系统：
  - Windows 98
  - Windows NT™
  - Windows 2000
  - Windows XP

图 **A-1.** 开发工具配置 **(1/2)**

(1) 当使用片上仿真 **QB-78K0LX3** 时



注　**1.** 从开发工具下载网站下载 78K0/LE3 的设备文件 (DF780495)
(http://www.necel.com/micro/ods/eng/index.html).

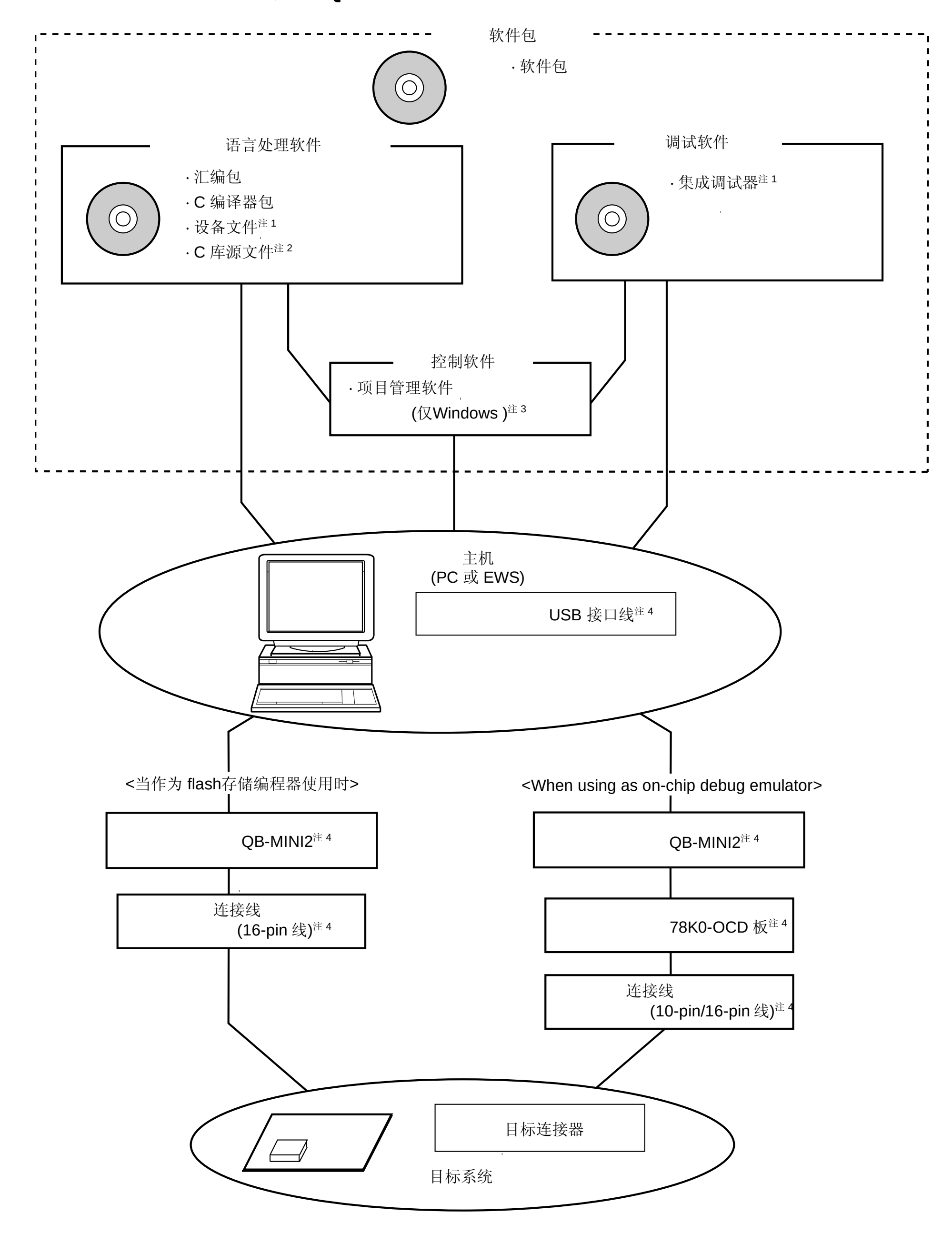
2. C 库文件源程序不包含在软件包内。

3. 项目管理软件 PM+包含在汇编包内。
PM+只能在 Windows 平台下操作.

4. QB-78K0LX3 提供集成调试器ID78K0-QB, USB接口电缆、电源线、片上调试仿真器QB-MINI2，连接线 (10-pin和 16-pin 线)和 78K0-OCD板. 其它产品均独立销售。
当使用QB-MINI2 时从开发工具下载网站下载用于QB-MINI2 操作的软件
(http://www.necel.com/micro/ods/eng/index.html) .

图 **A-1.** 开发工具配置 **(2/2)**

(2) 当使用具有编程功能的片上调试仿真器 **QB-MINI2** 时



注 **1.** 可从开发工具下载网站（http://www.necel.com/micro/ods/eng/index.html）下载 78K0/LE3（DF780495）的设备文件和集成调试器 (ID78K0-QB)。

2. C 库文件源程序不包含在软件包内.

3. 项目管理软件 PM+包含在汇编包内.
PM+ 仅用于 Windows

4. 购买片上调试仿真器QB-MINI2 时还提供USB接口电缆、连接电缆(10 引脚电缆和 16 引脚电缆)和 78K0-OCD板。其它产品均独立销售.
另外，可从MINICUBE2 的下载网站（http://www.necel.com/micro/en/development/asia/minicube2/minicube2.html）下载操作QB-MINI2 的软件。

A.1 软件包

SP78K0 78K0 系列软件包	开发工具（软件包）适用于 78K0 微控制器。
	部件号: μS××××SP78K0

备注　　××××型号随使用的主机和操作系统的不同而异。

μS××××SP78K0

××××	主机	操作系统	提供形式
AB17	PC-9800 系列	Windows (日文版)	CD-ROM
BB17	IBM PC/AT 兼容机	Windows (英文版)	

A.2 语言处理软件

RA78K0 汇编编译器	该汇编编译器将助记符编写的程序转换为微处理器可执行的目标代码。 该汇编编译器可以自动建立符号表和对传送指令进行优化。 该汇编编译器应当和设备文件（DF780495）一起使用。 **<在 PC 环境中使用 RA78K0 的注意事项>** 该汇编编译器是基于 DOS 应用的。 它也可以在 Windows 环境下使用，但必须在 Windows 环境下使用 Project Manager(包含在汇编编译器软件包中)。
	部件号: μS××××RA78K0
CC78K0 C 编译器	该编译器将 C 语言编写的程序转换为微控制器可执行的目标代码。 该编译器应当结合汇编编译器软件包和设备文件(两者均单独出售)一起使用。 **<在 PC 环境中使用 CC78K0 的注意事项>** 该 C 编译器是基于 DOS 应用的。 它也可以在 Windows 环境下使用，但必须在 Windows 环境下使用 Project Manager(包含在汇编编译器软件包中)。
	部件号: μS××××CC78K0
DF780495 注 1 设备文件	该文件包含设备特有的信息。 该设备文件应当结合工具(RA78K0、CC78K0、用于 78K0 的 SM+、ID78K0-QB) (均为独立销售)一起使用。 相应的 OS 和主机随使用工具而变化。
	部件号: μS××××DF780495
CC78K0-L 注 2 C 库源文件	该函数的源文件配置了目标库文件包含在 C 编辑包中. 本文件需要 C 编译包内的目标文件与用户规格相匹配.
	部件号: μS××××CC78K0-L

注　**1.**　DF780495 可以与RA78K0、CC78K0、用于 78K0 的SM+ 和ID78K0-QB一起使用。 从开发工具下载网站下载DF780495 (http://www.necel.com/micro/ods/eng/index.html).

2.　CC78K0-L 不包含软件包内 (SP78K0).

备注 ××××型号随使用的主机和操作系统的不同而异。

μS××××RA78K0
μS××××CC78K0
μS××××CC78K0-L

××××	主机	操作系统	提供形式
AB17	PC-9800 系列； IBM PC/AT 兼容机	Windows (日文版)	CD-ROM
BB17		Windows (英文版)	
3P17	HP9000 系列 700™	HP-UX™ (Rel. 10.10)	
3K17	SPARCstation™	SunOS™ (Rel. 4.1.4) Solaris™ (Rel. 2.5.1)	

μS××××DF780495

××××	主机	操作系统	提供形式
AB13	PC-9800 系列； IBM PC/AT 兼容机	Windows (日文版)	3.5-inch 2HD FD
BB13		Windows (英文版)	

A.3 控制软件

PM+ 项目管理器	这是一个控制软件，设计使得用户可以在 Windows 环境下有效地开发用户程序。所有用户开发过程中的操作，诸如启动编辑器、构建以及进行调试等，都可以在 PM+中进行。**<注意事项>** PM+ 包含在 C 编译包 RA780 之中。 它仅能在 Windows 下使用。

A.4 Flash存储器编程工具

A.4.1 当使用Flash存储器编程器 PG-FP5 和 FL-PR5 时

PG-FP5, FL-PR5 Flash 存储编程器	Flash 存储器编程器，专用于片上 Flash 存储器的微控制器.
FA-64GB-GAH-B FA-78F0465GB-GAH-RX FA-64GK-GAJ-B FA-78F0465GK-GAJ-RX Flash 存储器编程适配器	Flash 存储器编程适配器，用于连接到 Flash 存储器编程器使用。 • FA-64GB-GAH-B, FA-78F0465GB-GAH-RX: 用于 64pin 塑封 LQFP (GB-GAH type) • FA-64GK-GAJ-B, FA-78F0495GK-GAK-RX: 用于 64-pin 塑封 LQFP (GK-GAJ type)

备注 **1.** FL-PR5, FA-64GB-GAH-B, FA-78F0465GB-GAH-RX, FA-64GK-GAJ-B 和 FA-78F0465GK-GAJ-RX 是 Naito Densei Machida Mfg. Co., Ltd 的产品

TEL: +81-42-750-4172 Naito Densei Machida Mfg. Co., Ltd.

2. 使用 Flash 存储器编程适配器的最新版本。

A.4.2 当使用具有编程功能的片上调试仿真器QB-MINI2

QB-MINI2 具有编程功能的片上调试仿真器	这是具有片上 Flash 存储器的微控制器专用的 Flash 存储器编程器。 当使用 78K0/Lx3 微控制器开发应用系统时，它也可以当作片上调试仿真器，用于调试硬件和软件。 当作为 Flash 存储器编程器使用时，它应当和一个连接电缆(16 引脚电缆)和一个 USB 接口电缆一起使用，后者用于连接主机。
目标连接器规格	16 引脚通用电缆 (2.54 mm 宽)

备注 **1.** QB-MINI2 提供 USB 接口电缆、连接电缆(10 引脚电缆和 16 引脚电缆)和 78K0-OCD 板。其它产品均独立销售. 10 引脚电缆和 78K0-OCD 板 仅用于片上调试功能时。

2. 可从MINICUBE2 的下载网站（http://www.necel.com/micro/en/development/asia/minicube2/minicube2.html）下载操作QB-MINI2 的软件。

A.5 调试工具 (硬件)

A.5.1 当使用在线仿真器QB-78K0LX3 时

QB-78K0/Lx3 在线仿真器	当使用 78K0/Lx3 微处理器开发应用系统时，该在线仿真器用于调试硬件和软件。 它支持集成调试器(ID78K0-QB)。 该仿真器与电源和仿真探头，和用于连接此仿真器到主机的 USB 一起使用。 .
QB-144-CA-01 引脚检测适配器	该适配器用于示波器等的波形检测。
QB-80-EP-01T E 仿真探头	这是一种可以灵活使用的仿真探头，用于连接在线仿真器和目标系统。
QB-64GB-EA-09T, QB-64GK-EA-07T 交换适配器	该适配器用于实现从在线仿真器到目标连接器的引脚转换。 • QB-64GB-EA-09T: 用于 64 引脚塑封 LQFP (GB-GAH type) • QB-64GK-EA-07T: 用于 64 引脚塑封 LQFP (GK-GAJ type)
QB-64GB-YS-01T, QB-64GK-YS-01T 空间适配器	该适配器用于调整目标系统和在线仿真器之间的高度。 • QB-64GB-YS-01T: 用于 64 引脚塑封 LQFP (GB-GAH type) • QB-64GK-YS-01T: 用于 64 引脚塑封 LQFP (GK-GAJ type)
QB-64GB-YQ-01T, QB-64GK-YQ-01T YQ 连接器	该连接器用于连接目标连接器和交换适配器。 • QB-64GB-YQ-01T: 用于 64 引脚塑封 LQFP (GB-GAH type) • QB-64GK-YQ-01T: 用于 64 引脚塑封 LQFP (GK-GAJ type)
QB-64GB-HQ-01T, QB-64GK-HQ-01T 安装适配器	该适配器用于通过插座安装目标设备。 • QB-64GB-HQ-01T: 用于 64 引脚塑封 LQFP (GB-GAH type) • QB-64GK-HQ-01T: 用于 64 引脚塑封 LQFP (GK-GAJ type)
QB-64GB-NQ-01T, QB-64GK-NQ-01T 目标连接器	该目标连接器用于安装到目标系统上。 • QB-64GB-NQ-01T: 用于 64 引脚塑封 LQFP (GB-GAH type) • QB-64GK-NQ-01T: 用于 64 引脚塑封 LQFP (GK-GAJ type)

备注 **1.** QB-78K0LX3 提供一个电源和 USB 接口电缆。作为控制软件，还提供集成调试器 ID78K0-QB 和具有编程功能的片上调试仿真器 QB-MINI2，连接电缆(10 引脚电缆和 16 引脚电缆)和 78K0-OCD 板。
可从MINICUBE2 的下载网站（http://www.necel.com/micro/en/development/asia/minicube2/minicube2.html）下载操作QB-MINI2 的软件。

2. 封装内容根据产品型号而不同，如下所示。

封装内容 / 产品型号	在线仿真器	仿真探头	交换适配器	YQ连接器	目标连接器
QB-78K0LX3 -ZZZ	QB-78K0LX3	无			
QB-78K0LX3-T64GB		QB-80-EP-01T	QB-64GB-EA-09T	QB-64GB-YQ-01T	QB-64GB-NQ-01T
QB-78K0LX3-T64GK			QB-64GK-EA-07T	QB-64GK-YQ-01T	QB-64GK-NQ-01T

A.5.2 当使用具有编程功能的片上调试仿真器QB-MINI2 时

QB-MINI2 具有编程功能的片上调试仿真器	当使用 78K0/Lx3 微控制器开发应用系统时，该片上调试仿真器用于调试硬件和软件。 也可专门当作具有片上 Flash 存储器的微控制器的 Flash 存储器编程器使用。 当作为片上调试仿真器使用时，应当与一根连接电缆(10 引脚电缆和 16 引脚电缆)和 USB 接口电缆一起使用，后者用于连接主机和 78K0-OCD 板。
目标连接器规格	10 引脚通用电缆 (2.54 mm 宽)或 16 引脚通用电缆 (2.54 mm 宽)

备注 **1.** QB-MINI2 提供 USB 接口电缆、连接电缆(10 引脚电缆和 16 引脚电缆)和 78K0-OCD 板。其它产品均独立销售. 10 引脚电缆和 78K0-OCD 板 仅用于片上调试功能时。

2. 可从MINICUBE2 的下载网站（http://www.necel.com/micro/en/development/asia/minicube2/minicube2.html）下载操作QB-MINI2 的软件。

A.6 调试工具(软件)

ID78K0-QB 综合调试器	该调试器支持 78K0 微控制器的在线仿真器。 ID78K0-QB 是基于 Windows 的软件。 它改善了 C 兼容的调试功能，并使用集成窗口功能（结合源程序、分开显示和跟踪结果的存储器显示）来显示源程序的跟踪结果。 它应当结合设备文件一起使用。
	部件号: μS××××ID78K0-QB

备注 ××××型号随使用的主机和操作系统的不同而异。

μS××××ID78K0-QB

××××	主机	操作系统	提供形式
AB17	PC-9800 系列; IBM PC/AT 兼容机	Windows (日文版)	CD-ROM
BB17		Windows (英文版)	

附录 B 修订历史

B.1 本版本主要修订历史

(1/3)

页面	说明	分类
全部	• 增加 PG-FP5 和 FL-PR5 • 增加段按键扫描功能的解释	(b, c)
第一章 概述		
p. 17	增加在 **1.1 特征**中 ROM, RAM 容量的表中**注 2**	(d)
pp. 20 to 22	增加**备注 2** 到 **1.4 引脚配置 (俯视图)**	(c)
pp. 25 to 27	增加**注** 到 **1.5 78K0/Lx3 微控制器系列介绍**的功能列表中	(c)
p. 30	增加**注 4** 到 **1.7 功能概述 (μPD78F044x)**	(c)
p. 33	增加**注 4** 到 **1.8 功能概述 (μPD78F045x)**	(c)
p. 36	增加**注 5** 到 **1.9 功能概述 (μPD78F046x)**	(c)
第三章 CPU 架构		
pp. 56, 58, 60, 62, 64	增加**注 3** 到 **图 3-2**, **图 3-4**, **图 3-6**, **图 3-8** 和 **图 3-10 的存储空间**	(b)
pp. 70, 72, 74, 76, 78	增加**注** 到**图 3-12**, **图 3-14**, **图 3-16**, **图 3-18**, **图 3-20 数据存储空间与寻址方式的对应关系**	(b)
p. 95	更改 **3.3.3 表间接寻址**的说明	(c)
第四章 端口功能		
p. 110	更改 **图 4-8.P20 ~ P27 的框图**	(a)
p. 129	更改 **图 4-28. 端口寄存器格式** 和增加**注 1, 2** 和 **3**	(b, c)
p. 130	增加**注** 到 **图 4-29. 上拉电阻选择寄存器的格式**	(c)
第五章 时钟发生器		
p. 151	更改 **图 5-9. 内部高速振荡微调寄存器（HIOTRM）的格式**	(b)
p. 157	更改 **图 5-13. 打开供电电源上电时的时钟发生器操作**	(b)
p. 158	更改 **图 5-14. 当打开供电电源上电时的时钟发生器操作**的**注意事项**	(b)
第六章 16 位定时器/事件计数器 00		
p. 188	增加**注 2** 到 **图 6-10. 输入切换控制寄存器（ISC）的格式**	(c)
p. 223	修改**图 6-42. (f) 16 位 捕获/比较寄存器 000（CR000）**	(a)
p. 250	增加 **6.6 (12) 16 位定时器计数器 00（TM00）的读取**	(c)
第七章 8 位定时器/事件计数器 50、51 和 52		
p. 272	增加 **7.5 (3) 8 位定时器计数器 5n（TM5n）的读取**	(c)
第八章 8 位定时器 H0、H1 和 H2		
p. 275	更改 **图 8-2. 8 位定时器 H1 的框图**	(a)
p. 279	更改 **图 8-6. 8 位定时器 H 模式寄存器 0（TMHMD0）的格式**	(c)
p. 301	增加 **8.4.4 由定时器 51 计数器控制载波时钟的数量**	(b, c)

备注 下面显示的是上表中“分类” 的类别.

(a): 错误更改, (b): 增加/更改规格, (c): 增加/更改描述或注, (d): 增加/更改封装,部件号码或管理部门,
(e): 增加/更改相关文档

(2/3)

页面	说明	分类
第九章 实时计数器		
p. 307	更改图 **9-4.** 实时计数器控制寄存器 **1**（**RTCC1**）的格式的注意事项	(b, c)
p. 311	更改 表 **9-2.** 显示的时间数字	(c)
p. 313	增加注意事项到 图 **9-11.** 星期计数寄存器（**WEEK**）的格式	(c)
p. 315	更改 **(13)** 钟表计数寄存器（**SUBCUD**）	(c)
p. 316	增加注释到 **(15)** 闹钟小时寄存器（**ALARMWH**）	(c)
p. 318	增加注 到 图 **9-18.** 实时计数器的启动操作流程图	(c)
p. 319	增加 **9.4.2** 启动操作后转到 **STOP** 模式	(c)
p. 323	增加 **9.4.5** 实时计数器的 **1 Hz** 输出	(c)
p. 323	增加 **9.4.6** 实时计数器的 **32.768 kHz** 输出	(c)
p. 324	增加 **9.4.7** 实时计数器的 **512 Hz**，**16.384 kHz** 输出	(c)
p. 325	增加 **9.4.8** 实时计数器的钟表误差修正的示例	(c)
第十章 看门狗定时器		
p. 336	更改 **10.4.3** 设置看门狗定时器的窗口打开时期的备注	(a)
第十二章 **10** 位逐次逼近型 **A/D** 转换器**(**仅限μ**PD78F045x** 和 **78F046x)**		
p. 345	更改表 **12-2. A/D** 转换时间的选择的注意事项 **1**	(c)
p. 361	更改 **12.6 (11)** 内部等效电路	(b)
第十三章 **16** 位 **ΔΣ** 型 **A/D** 转换器（仅限 **μPD78F046x**）		
pp. 362 ~385	本章节全部更改	(b, c)
第十四章 串行接口 **UART0**		
p. 405	更改 表 **14-5.** 波特率发生器的数据设置	(b)
第十七章 **LCD** 控制器**/**驱动器		
pp. 465 ~526	本章节全部更改	(b, c)
第十九章 遥控器接收器		
pp. 552 ~583	本章节全部更改	(b, c)
第二十章 中断功能		
p. 591	更改 图 **20-2.** 中断请求标志寄存器（**IF0L**，**IF0H**，**IF1L**，**IF1H**）的格式	(a)
p. 593	更改 图 **20-3.** 中断屏蔽标志寄存器（**MK0L**，**MK0H**，**MK1L**，**MK1H**）的格式	(a)
p. 594	更改 图 **20-4.** 优先级指定标志寄存器（**PR0L**，**PR0H**，**PR1L**，**PR1H**）的格式	(a)
第二十一章 按键中断功能		
p. 605	增加注意事项 **5** 到图 **21-2.** 按键返回模式寄存器 （**KRM**） 的格式	(c)
第二十二章 待机模式		
p. 613	更改 图 **22-4.** 通过复位释放 **HALT** 模式	(c)
pp. 615, 616	更改 表 **22-3. STOP** 模式下的操作状态**,** 注 **2** 和 注意事项 **3** 及增加注意事项 **4**	(c)
p. 619	更改 图 **22-7.** 由复位释放 **STOP** 模式	(c)

备注 下面显示的是上表中“分类” 的类别.

(a): 错误更改, (b): 增加/更改规格, (c): 增加/更改描述或注, (d): 增加/更改封装,部件号码或管理部门,
(e): 增加/更改相关文档

(3/3)

页面	说明	分类
第二十三章 复位功能		
p. 622	更改 图 **23-2.** 由 **RESET** 输入进行复位的时序	(c)
p. 622	更改 图 **23-3.** 由于看门狗定时器溢出进行复位的时序	(c)
p. 623	更改 图 **23-4.** 在 **STOP** 模式下通过 **RESET** 输入进行复位的时序	(c)
第二十四章 上电清零电路		
p. 632	更改 图 **24-2.** 由上电清零电路和低电压检测器产生内部复位信号的时序**(1)**	(b)
p. 633	更改图 **24-2.** 由上电清零电路和低电压检测器产生内部复位信号的时序**(2)** 注意事项 **2**	(b, c)
第二十七章 **FLASH** 存储器		
p. 661	更改 注 **2** 和 增加注意事项 到 表 **27-3. 78K0/LE3** 和专用 **Flash** 存储器编程器之间的连线	(c)
p. 665	更改 in 表 **27-4.** 引脚连接中注 **1**	(c)
p. 668	更改 **27.6.6** 其它信号引脚的注意事项 **2**	(c)
p. 669	更改 图 **27-12. Flash** 存储器操作过程	(b)
p. 670	更改 表 **27-7.** 通信模式	(b)
pp. 674~ 676	增加 **27.9** 使用 **PG-FP5** 时各个命令的处理时间（参考）	(b, c)
pp. 677 ~ 686	更改 **27.10** 通过自编程进行 **Flash** 存储器编程	(b, c)
第二十九章 片上调试功能		
pp. 731, 732	本章节全部更改	(c)
第三十一章 电气特性（标准产品）		
p. 746	增加注意事项	(c)
p. 750	增加推荐振荡器常数	(b)
p. 754	更改 供应电流值 和 注 **1, 6** 和 增加注 **5** 到 **DC** 特性	(b)
p. 755	更改 **DC** 特性 **ΔΣ** 型 **A/D** 转换器操作电流	(b)
p. 765	更改 **16** 位 **ΔΣ** 型 **A/D** 转换器特性	(b)
p. 766	更改 **LCD** 特性的注 **3**	(b)
p. 769	更改 **Flash** 存储器编程特性和增加注 **1, 4**	(b, c)
第三十三章 推荐焊接条件		
p. 772	增加章节	(b, c)
附录 **A** 开发工具		
pp. 775 ~ 782	增加附录 A	(c)

备注 下面显示的是上表中“分类” 的类别.

(a): 错误更改, (b): 增加/更改规格, (c): 增加/更改描述或注, (d): 增加/更改封装,部件号码或管理部门,
(e): 增加/更改相关文档

详细信息请联系：

中国区

MCU 技术支持热线：
电话：+86-400-700-0606 (普通话)
服务时间：9:00-12:00，13:00-17:00 （不含法定节假日）

网址：
http://www.cn.necel.com/ （中文）
http://www.necel.com/ （英文）

[北京]

日电电子（中国）有限公司
中国北京市海淀区知春路 27 号
量子芯座 7，8，9，15 层
电话：（+86）10-8235-1155
传真：（+86）10-8235-7679

[上海]

日电电子（中国）有限公司上海分公司
中国上海市浦东新区银城中路 200 号
中银大厦 2409-2412 和 2509-2510 室
电话：（+86）21-5888-5400
传真：（+86）21-5888-5230

上海恩益禧电子国际贸易有限公司
中国上海市浦东新区银城中路 200 号
中银大厦 2511-2512 室
电话：（+86）21-5888-5400
传真：（+86）21-5888-5230

[长春]

日电电子（中国）有限公司长春分公司
吉林省长春市朝阳区
西安大路 727 号中银大厦 A 座 1609 室
电话：(+86)431-8859-7533 / 8859-8533
传真：(+86)431-8680-2944

[深圳]

日电电子（中国）有限公司深圳分公司
深圳市福田区益田路卓越时代广场大厦 39 楼
3901，3902，3909 室
电话：（+86）755-8282-9800
传真：（+86）755-8282-9899

[香港]

香港日电电子有限公司
香港九龙旺角太子道西 193 号新世纪广场
第 2 座 16 楼 1601-1613 室
电话：（+852）2886-9318
传真：（+852）2886-9022
2886-9044

[成都]

日电电子（中国）有限公司成都分公司
四川省成都市二环路南三段 15 号
天华大厦 608 室
电话：(+86)28-8512-5224
传真：(+86)28-8512-5334

[大连]

日电电子（中国）有限公司长春分公司
大连市中山路 88 号天安国际大厦 2701 室
电话：(+86)411-8230-8815 / 8230-8825
传真：(+86)411-8230-8835